SUPER 10

Anno 2
n. 10 - Novembre 1985
L. 7.500
Spedizione in abbonamento postale Gruppo III/70

COMMODORE 64

SUPERCOMMODORE
la rivista per gli utenti dei prodotti COMMODORE

Una pubblicazione della
J.soft EDITRICE
in collaborazione con
GRUPPO EDITORIALE JACKSON



RETTANGOLI

COME RIALLINEARE LA TESTINA DEL DRIVE 1541

PAINTBOX

ULTRAWEDGE

RIDUTTORE E VELOCIZZATORE DI PROGRAMMI

FILE PROTECTOR

COSTRUISCI LA PRIMA BIBLIOTECA COMPLET
DI INFORMATICA PERSONALE

QUADERNI JACKSON
DI PERSONAL COMPUTER



Volume per volume, settimana per settimana, tutti gli argomenti dell'informatica personale: da cosa significa "vivere col computer" al BASIC; dal COBOL al PASCAL e ai più diffusi linguaggi e sistemi operativi (MS-DOS, CP/M, MSX ...) Dall'hardware al software, dalla programmazione dei personal computer all'utilizzo dei più affermati pacchetti software: Multiplan, Visicalc, Lotus 1-2-3, Symphony dBase II, Wordstar ...



PER CHI INIZIA

la sua avventura nel mondo dei computer e ha bisogno di una guida sicura per costruire le basi della sua conoscenza ed esperienza.



PER CHI DESIDERA APPROFONDIRE

per ragioni di studio, hobby, lavoro, precise tematiche di software, di hardware e di programmazione

PER CHI VUOLE POSSEDERE

per esigenze professionali o culturali, un'organica Biblioteca di testi di base, scritti da esperti qualificati, coerente con un disegno editoriale di ampio respiro.



IN EDICOLA I PRIMI 3 LIBRI A SOLE L. 9.800

QUADERNI JACKSON
DI PERSONAL COMPUTER

Una Biblioteca di agili monografie, rivolta ad un vastissimo pubblico: appassionati, professionisti, studenti e tecnici.
30 Volumi pratici redatti con uno stile chiaro, competente, accurato ed essenziale.



GRUPPO EDITORIALE JACKSON

DIVISIONE LIBRI

SUPER COMMODORE 64

J. *soft* s.r.l.
DIREZIONE, REDAZIONE, AMMINISTRAZIONE
Viale Restelli, 5
20124 MILANO
Tel. (02) 68.88.228

DIRETTORE RESPONSABILE:
Pietro Dell'Orco

COORDINAMENTO TECNICO:
Riccardo Paolillo

REDAZIONE:
Mauro Cristuib Grizzi

HANNO COLLABORATO A QUESTO NUMERO
S. Colombo
L. Priotto

GRAFICA E IMPAGINAZIONE:
Margherita La Noce
Ivana Rossi
Raffaella Toffolatti

FOTOCOMPOSIZIONE:
Graphotek - Via Astesani, 16
Tel. 02/64.80.397
20161 MILANO

CONTABILITÀ:
Giulia Pedrazzini
Flavia Bonaiti

UFFICIO ABBONAMENTI:
Gianna Dell'Olio

AUTORIZZAZIONE ALLA PUBBLICAZIONE:
Tribunale di Milano n° 201
del 14.04.1984

STAMPA:
Litografia del Sole
Albairate (MI)

PUBBLICITÀ
Concessionaria per l'Italia e l'Estero J. Advertising s.r.l.
Viale Restelli, 5
20124 MILANO
Tel. (02) 68.82.895-68.80.606-68.87.233
Tlx. 316213 REINA I

Concessionaria esclusiva per la DIFFUSIONE in Italia ed Estero:
SODIP - Via Zuretti, 25
20125 MILANO

Spedizione in abbonamento postale Gruppo III/70

Prezzo della rivista L. 7.500
Numero arretrato L. 15.000
Abbonamento annuo: L. 82.500 (11 numeri con cassetta);
L. 38.500 (11 numeri senza cassetta);
per l'estero: L. 110.000 (11 numeri con cassetta) - L. 50.000 (11 numeri senza cassetta)

I versamenti devono essere indirizzati a:
J. *soft* s.r.l.
Viale Restelli, 5 - 20124 MILANO
mediante emissione di assegno bancario, vaglia o utilizzando il c/c postale n. 19445204.
Per i cambi di indirizzo indicare, oltre al nuovo, anche l'indirizzo precedente ed allegare alla comunicazione l'importo di L. 500, anche in francobolli

GLI ARTICOLI TRADOTTI SONO TRATTI DALLA RIVISTA COMPUTE! O DA COMPUTE! GAZZETTE - COMPUTE! PUBLICATION, INC.
P.O. BOX 5406,
GREENSBORO,
NC 27403 - USA



Per collaborare a SUPERCOMMODORE

La rivista è interessata ad articoli e programmi riguardanti la linea Commodore, di interesse generale, chiari ed esaurienti. Vi proponiamo una piccola "guida", che ha particolarmente lo scopo di rendere più accurata la stesura dei vostri manoscritti: seguendo i nostri consigli si accresceranno le probabilità che le vostre buone idee e i vostri programmi vengano pubblicati.

1 l'angolo superiore sinistro della prima pagina dovrà contenere: nome, cognome, indirizzo, numero telefonico, codice fiscale, data di spedizione, luogo e data di nascita.

2 l'angolo superiore destro della prima pagina dovrà contenere la marca e il tipo di computer al quale il lavoro si riferisce, unitamente ad eventuali espansioni di memoria o periferiche richieste.

3 il titolo dell'articolo, sottolineato, dovrà iniziare a circa due terzi in altezza della prima pagina.

4 le pagine seguenti potranno essere battute normalmente, con la condizione che l'angolo superiore destro contenga un'abbreviazione del titolo e del cognome, unitamente al numero di pagina. Per esempio, Sprite Ed.../Brambilla/2.

5 il testo dell'articolo dovrà essere battuto a macchina con interlinea di uno spazio e mezzo, massimo due spazi; un margine di almeno un centimetro dovrà essere lasciato su entrambi i lati dello scritto.

6 dovranno essere usati fogli in formato UNI A4 (cm. 21 x 29,7) e il testo, scritto in caratteri maiuscoli e minuscoli, dovrà occupare una sola facciata del foglio.

7 nel caso il testo comprenda più fogli, questi dovranno essere uniti con un fermaglio o con un punto metallico aperto.

8 avendo intenzione di spedire più di un articolo, questi dovranno essere inviati separatamente insieme alla rispettiva copia su supporto magnetico.

9 programmi brevi (meno di 20 linee) potranno essere inseriti nel testo, mentre programmi più lunghi dovranno essere listati separatamente. È **ESSENZIALE** per noi disporre di una copia del programma registrata più volte su supporto magnetico, su entrambi i lati dello stesso. È preferibile usare nastri di buona qualità e di lunghezza non eccessiva; la cassetta o il disco dovranno essere etichettati con il nome dell'autore, il titolo dell'articolo, il computer interessato e soprattutto le eventuali espansioni richieste. Come suggerimenti di programmazione si consiglia di usare le istruzioni "CHR$(x)", "TAB(x)", SPC(x)", piuttosto che stringhe di manipolazione del cursore.
Ad esempio, per uno scroll di 5 linee l'istruzione "FORI = 1TO5:PRINT:NEXT". è molto più interpretabile di 5 Q inverse; e, invece di una dozzina di simboli di cursore a destra, perché non usare semplicemente "PRINT SPC (12)"? Un rapido controllo dei programmi per operare queste sostituzioni sarà molto apprezzato da noi e dai lettori.

10 per maggior chiarezza, all'interno dell'articolo è conveniente usare caratteri maiuscoli riferendosi a istruzioni BASIC (esempio RETURN, LIST, RND, PRINT ecc.). Se si desidera evidenziare una parola, è preferibile sottolinearla piuttosto che scriverla in carattere maiuscolo.

11 gli articoli ed i programmi potranno avere qualsiasi lunghezza: da una routine di una sola linea fino a programmi molto complessi.

12 volendo includere diapositive, queste dovranno avere formato 24×36, o 6×6.

13 non prenderemo in considerazione articoli che siano stati sottoposti ad altre case editrici.

14 il materiale non pubblicato non verrà restituito.

15 il compenso per la collaborazione prestata sarà commisurato alla complessità e all'interesse del testo e/o del programma (da un minimo di L. **50.000** a un massimo di L. **300.000**). Il pagamento verrà effettuato in caso di pubblicazione del lavoro.

Inviate idee e programmi a:
SUPERCOMMODORE
Viale Restelli, 5
20124 Milano

e saremo lietissimi di pubblicare i contributi migliori.

La Redazione

OK-LIST

Sappiamo tutti quanto sia difficile digitare correttamente un programma (ovviamente, se non si dispone della cassetta allegata a SUPERCOMMODORE!). Errori di battitura anche banali possono provocare il malfunzionamento del programma stesso, o anche il blocco completo del sistema, obbligandoci a rifare il lavoro.
Ma il punto di forza dei calcolatori è che essi eccellono nei compiti noiosi e che richiedono precisione. Scoprire un errore nella battitura di un listato è appunto un compito noioso, quindi perché non sfruttare il tuo computer per controllare i listati da te digitati?
È proprio questo lo scopo di OK-LIST, il programma che ti permetterà di risparmiare tempo e fatica. Annidato all'interno del tuo computer, OK-LIST controlla automaticamente ogni riga battuta, mostrandoti un numero in cima allo schermo ogni volta che premi il tasto RETURN. Un numero corrispondente nel listato del programma pubblicato ti permette il confronto, ed uno sguardo è sufficiente per avere la conferma di aver digitato la linea correttamente.
OK-LIST è un breve programma in linguaggio macchina residente nel buffer del registratore. Esso rimane in tale zona finché spegni il computer o usi l'unità a cassette.

Come digitare OK-LIST

Digita il listato che segue, facendo attenzione a non commettere errori (per fortuna il listato è molto breve!). Dopo aver finito salvalo su disco o nastro, per lo meno un paio di volte, prima di mandarlo in esecuzione con il comando RUN, perché il programma BASIC, dopo aver memorizzato la routine in linguaggio macchina (le istruzioni DATA), si autocancella.

Come si usa OK-LIST

A questo punto puoi iniziare a battere il tuo primo listato "a prova di errore". Se non hai OK-LIST in memoria, caricalo e attivalo con RUN.
Digita quindi la prima linea di programma: come puoi vedere, in fondo ad essa c'è la parola ":rem" seguita da un numero, che si chiama checksum.
NON COPIARE QUESTA PARTE DELLA LINEA! Ora premi il tasto RETURN, come al solito. In alto a sinistra apparirà il checksum; confrontalo con quello che vedi scritto sulla rivista dopo ":rem". Se i due numeri sono uguali, vuol dire che la linea è stata digitata correttamente, altrimenti controllala, perché evidentemente hai commesso un errore. Facile, no?
OK-LIST non è pignolo per quel che riguarda gli spazi bianchi, dato che non fa caso alla loro presenza. Ciò torna a tuo vantaggio, dal momento che la spaziatura non è, in genere, importante. Ma all'interno delle istruzioni PRINT gli spazi spesso si rivelano necessari, quindi presta attenzione quando digiti questo tipo di istruzione. Un'altra cosa a cui devi badare è che usando le forme abbreviate dei comandi (ad esempio ? per PRINT) il checksum visualizzato non coinciderà con quello presente sul listato. Ciò è tuttavia risolvibile, premendo il tasto RETURN su una linea dopo averla listata: le abbreviazioni saranno ricodificate nella loro forma intera e quindi OK-LIST sarà in grado di svolgere perfettamente la sua funzione. In questo modo, inoltre, avrai la possibilità di controllare interi gruppi di righe in una volta sola.
Ovviamente, nessuno è perfetto, ed anche OK-LIST ha qualche limite: se per caso ti capitasse (ma è un errore poco frequente) di "anagrammare" durante la digitazione una linea (per esempio invece di battere 10 PRINT"VAI", digiti per errore 10 PRINT"VIA"), OK-LIST non se ne accorgerebbe. Inoltre la presenza del carattere SHIFT-SPAZIO modifica il numero di checksum rispetto al carattere SPAZIO normale, anche se la differenza non risulta visibile sul video del tuo computer.
Se intendi sospendere il lavoro di digitazione – o lo hai terminato – puoi salvare su nastro o disco il tuo programma: ricordati comunque di premere RUN/STOP e RESTORE prima di dare il comando SAVE. Al momento di riprendere il lavoro, tuttavia, non potrai più utilizzare OK-LIST, dal momento che qualsiasi operazione su cassetta lo distrugge.
Se vuoi disabilitare OK-LIST in qualunque momento, senza perdere il tuo programma in memoria, premi RUN/STOP e RESTORE. Per riattivarlo batti:

SYS 886
Buon lavoro!

```
10 PRINT"{CLR}ATTENDERE PREGO..."
11 FORI=886TO1018:READA:CK=CK+A:POKEI,A:N
   EXT
12 IF CK<>17539 THEN PRINT"{GIU'}ERRORE N
   ELLE ISTRUZIONI DATA":END
13 SYS886:PRINT"{CLR}{ 2 GIU'}OK-LIST ATT
   IVATO.":NEW
14 DATA173,036,003,201,150,208
15 DATA001,096,141,151,003,173
16 DATA037,003,141,152,003,169
17 DATA150,141,036,003,169,003
18 DATA141,037,003,169,000,133
19 DATA254,096,032,087,241,133
20 DATA251,134,252,132,253,008
21 DATA201,013,240,017,201,032
22 DATA240,005,024,101,254,133
23 DATA254,165,251,166,252,164
24 DATA253,040,096,169,013,032
25 DATA210,255,165,214,141,251
26 DATA003,206,251,003,169,000
27 DATA133,216,169,019,032,210
28 DATA255,169,018,032,210,255
29 DATA169,058,032,210,255,166
30 DATA254,169,000,133,254,172
31 DATA151,003,192,087,208,006
32 DATA032,205,189,076,235,003
33 DATA032,205,221,169,032,032
34 DATA210,255,032,210,255,173
35 DATA251,003,133,214,076,173
36 DATA003
```



READ &WRITE

La posta dei lettori

Tutti in riga

Essendo un possessore di C64, spesso mi sono imbattuto in un problema che ho cercato di risolvere con lunghe ed estenuanti righe di programma, con tutti i rischi di errore connessi. Mi riferisco all'incolonnamento delle cifre a destra anziché a sinistra.
Vi sarei grato se poteste pubblicare sulla rivista una vostra risposta circa la possibilità o meno di risolvere facilmente questo problema. **Paolo Belardi**

R. Purtroppo il BASIC del C64 è molto limitato, e non possiede forme di istruzioni PRINT tali da ottenere direttamente l'incolonnamento dei numeri a destra, oppure delle colonne di cifre allineate al punto decimale.
Il problema è tuttavia facilmente risolvibile, se si fa ricorso alla locazione 211, che controlla la colonna in cui viene posizionato il cursore prima di un'istruzione PRINT.
I due programmini che seguono stampano rispettivamente una colonna di numeri allineata a destra e una colonna di numeri allineata al punto decimale, leggendo i numeri da stampare in una istruzione DATA. Le modifiche da apportare per inserirli nei propri programmi come subroutine sono veramente minime e alla portata di tutti.

```
20 INPUT"COLONNA";C:PRINT"(CLR)
   "
30 FORI=1TO10:READD(I):NEXT
40 FORI=1TO10
50 K=C-LEN(STR$(D(I)))
60 POKE211,K
70 PRINTD(I)
80 NEXT
90 DATA10,100,1000,12.3,2.5,1,1
   23,9,8.76,32.9
```

```
20 INPUT"COLONNA";C:PRINT"(CLR)
   "
30 FORI=1TO10:READD(I):NEXT
40 FORI=1TO10
50 K=C-LEN(STR$(INT(D(I))))
60 POKE211,K
70 PRINTD(I)
80 NEXT
90 DATA10,100,1000,12.3,2.5,1,1
   23,9,8.76,32.9
```

Per un facile "debug"

Digitando molti listati della vostra e di altre riviste, mi capita sovente, dando il RUN al programma, di ottenere un messaggio di "?ILLEGAL QUANTITY ERROR IN ...". Listando la linea incriminata, rilevo che si tratta sempre di una linea contenente un'istruzione READ ed una POKE. So che l'errore è dovuto a un dato errato letto con la READ, dato che una volta immesso in memoria con la POKE genera l'errore. Il problema è che il messaggio d'errore fa riferimento alla linea contenente la READ e la POKE, e non alla linea DATA contenente il numero errato.
Soprattutto quando nel listato ci sono decine e decine di linee DATA, contenenti ognuna una decina di numeri, il lavoro di verifica risulta assai gravoso.
Potreste cortesemente indicare se esiste qualche sistema per risalire in qualche modo alla linea DATA contenente l'errore? **Vincenzo De Marchi**

R. Come giustamente afferma, un messaggio di "?ILLEGAL QUANTITY ERROR IN ...", generato da una linea che legge (READ) dei numeri in istruzioni DATA e li inserisce in memoria con una istruzione POKE, dipende da un numero errato in una delle istruzioni DATA.
I numeri che possono essere "pokati" in memoria possono infatti assumere soltanto valori compresi nell'intervallo tra 0 e 255, pena il famoso messaggio d'errore. Se una linea DATA, per un errore di digitazione, contiene ad esempio un numero "invertito" (come 525 invece di 255), questo errore non viene rilevato da OK-LIST in fase di battitura e genera il messaggio di cui sopra, in corrispondenza della linea che effettua la POKE. Come fare per individuare a colpo sicuro la linea DATA "incriminata"?
Fortunatamente ci viene in aiuto una coppia di puntatori nella memoria del computer, precisamente le locazioni 63 e 64 di pagina zero, che contengono il numero dell'ultima linea DATA letta dal computer. Dopo che è apparso il messaggio d'errore sarà quindi sufficiente digitare in modo diretto:

PRINTPEEK (63) + 256*PEEK (64)

Il numero che verrà stampato corrisponde all'ultima linea DATA letta dal computer, prima che l'esecuzione del programma fosse interrotta dal messaggio d'errore: sarà proprio lì che si "annida" il numero errato. Il programmino che segue esemplifica quanto descritto, leggendo in linea 10 i numeri contenuti nelle linee DATA ed inserendoli in memoria con una POKE. Dal momento che il valore contenuto in linea 70 è chiaramente fuori intervallo per la POKE, il programma si fermerà con un "?ILLEGAL QUANTITY ERROR IN 10". A questo punto basterà digitare in modo diretto GOTO2 per avere, a dimostrazione, il numero della linea DATA che contiene il valore errato.

```
1 GOTO10
2 PRINT"NUMERO LINEA DATA:"PEEK
  (63)+256*PEEK(64):END
10 FORI=49152TO49152+10:READD:P
   OKEI,D:NEXT
20 DATA1
30 DATA2
40 DATA3
50 DATA4
60 DATA5
70 DATA257
80 DATA7
90 DATA8
100 DATA9
110 DATA10
```

Una serie di quesiti

Spettabile Redazione,
ho una serie di quesiti da porvi:
1. Sono molto preoccupato per il C64: verrà sostituito dal C128, come il VIC 20 è stato rimpiazzato dal C16, ed uscirà di produzione?
2. Perché i programmi della cassetta allegata alla rivista non sono "superturboregistrati", come annunciato sul numero 3 di SUPERCOMMODORE?
3. Ho grossi problemi con il registratore: nonostante esso carichi perfettamente

tutti i programmi ed i giochi che acquisto, riesco a registrare una percentuale incredibilmente piccola dei giochi che copio o che programmo io stesso! Cosa posso fare? Ho notato però che questa percentuale varia a seconda del tipo di cassette da me usato... dipende da questo? E se sì, quali cassette devo usare?
4. Perché anche quando riesco a registrare un programma il mio C64 molto frequentemente lo salva con nomi incredibilmente strani e del tutto diversi da quelli da me scelti?

Tullio Tinti

R. Rispondiamo con ordine a questa serie di quesiti:
1. La Commodore non ha affatto annunciato di voler ritirare dalla produzione il C64, né del resto una simile scelta potrebbe essere indotta dal mercato di questa macchina, che, come sempre, "tira bene". Lo stesso C128 contiene entro di sé un C64, per poter far "girare" l'enorme massa di software esistente per il C64 stesso. Mentre il VIC 20 era un computer ormai superato (vedi schermo a 22 colonne e pochissima memoria), il C64 è tuttora uno dei migliori home computer disponibili sul mercato. Non crediamo quindi che ci sia da preoccuparsi a breve termine per la "vita" del C64.
2. Contrariamente a quanto annunciato sul numero di Marzo, i programmi contenuti sulle cassette allegate a SUPERCOMMODORE non sono in formato "Superturbo".
Abbiamo fatto questa scelta in dipendenza del fatto che i programmi registrati in turbo sono meno tolleranti rispetto ad eventuali non perfette tarature dei registratori, e quindi avrebbero dato, anche se ad un numero esiguo di lettori, più svantaggi che vantaggi. Nulla comunque vieta di duplicare "in Superturbo" i nostri programmi su un'altra cassetta, tanto più che quelli in linguaggio macchina sono sempre corredati dei rispettivi indirizzi iniziale e finale.
3. Il tipo di inconveniente, da come descritto, sembra possa effettivamente derivare dall'uso di un tipo di nastro magnetico non adatto ai registratori Commodore. Questi ultimi non sono infatti dotati di regolazione del BIAS e dell'equalizzazione, e lavorano quindi al meglio con nastri di qualità standard (BIAS basso ed equalizzazione a 120 microsecondi. Cassette di qualità superiore, come quelle al cromo o metal, se usate con un normale registratore Commodore, ne peggiorano soltanto l'affidabilità. Una cassetta di qualità standard e di buona marca, oltre a costare meno, garantisce risultati migliori di qualsiasi cassetta al cromo o metal.
4. Perché in fase di registrazione vengono perduti o deformati alcuni byte nella testata (header) del programma. È un inconveniente molto probabilmente legato a quanto descritto nel punto precedente.

Piccola guida per l'input dei programmi

Molti dei programmi listati da **SUPERCOMMODORE** contengono particolari caratteri di controllo (controllo cursore, tasti colore, video-inverse, ecc.).
Per rendere chiaro ciò che dev'essere battuto quando è necessario inserire uno dei suddetti caratteri sono state stabilite le seguenti convenzioni.
Generalmente i listati contengono delle «parole» racchiuse tra parentesi graffe { }; tali parole rappresentano particolari caratteri di controllo: {**GIÙ**} significa premere il simbolo del cursore verso il basso. {**5 SPAZI**} vuol dire battere 5 volte la barra-spazio.
Per indicare che un tasto dev'essere «shiftato», cioè premuto insieme al tasto SHIFT, nel listato il simbolo di questo tasto sarà sottolineato.
Per esempio S significa che dev'essere premuto il tasto S mentre è abbassato il tasto SHIFT.
La manovra farà apparire sullo schermo un piccolo «cuore».
Trovando un simbolo sottolineato chiuso tra parentesi graffe (esempio {**10 N**}) dovrà essere interpretato come «premere il carattere indicato per il numero di volte che lo precede nelle parentesi»: nel nostro esempio premere 10 volte il tasto N «shiftato».
Se il tasto da premere è indicato tra le parentesi [⟨⟩], vuol dire che lo stesso dovrà essere premuto mentre è premuto il tasto «Commodore» (il tasto «Commodore» è quello posto nell'angolo in basso a sinistra).
Se il tasto scritto tra [⟨⟩] è preceduto da un numero, ciò significa che il tasto dev'essere premuto per il numero di volte indicato.
Raramente si potrà trovare un carattere alfabetico solitario racchiuso tra parentesi graffe. Tale carattere sul C64 può essere battuto mentre è premuto il tasto CTRL.
Ad esempio {**A**} sta ad indicare la sequenza CTRL-A.
A proposito del «modo virgolette» è nota la possibilità di muovere il cursore sullo schermo con i tasti CRSR. Ogni tanto i programmatori desiderano muovere il cursore durante l'esecuzione del programma. È per questo motivo che nei programmi si troveranno dei {**SIN**}, {**HOME**} e {**BLU**}.
L'unico modo perché il calcolatore distingua il comando diretto da quello inserito nel programma è il «modo virgolette». Premendo il tasto «virgolette» (tasti SHIFT 2) il calcolatore si predispone appunto in «modo virgolette».
Battendo un carattere qualsiasi, cercando poi di correggerlo muovendo il cursore, si otterrà solo un tratto verticale in un quadratino inverso. Questo è infatti il simbolo del cursore a sinistra nel «modo virgolette».
L'unico comando di editing non utilizzabile all'interno di un programma è il DEL. Battendo nuovamente le «virgolette» il calcolatore lascia il «modo virgolette». Si può accedere al «modo virgolette» quando inserite spazi nella linea.
In ogni caso, il sistema più semplice per uscire dal «modo virgolette» è quello di premere il tasto RETURN.
Utilizzare la tabella che segue quando è necessario inserire comandi relativi al cursore e ai tasti colore.

Quando leggete:	Premete:	Vedrete:	Quando leggete:	Premete:	Vedrete:	Quando leggete:	Premete:	Vedrete:
{CLR}	SHIFT CLR/HOME		{CYN}	CTRL 4		[<7>]	C= 7	
{HOME}	CLR/HOME		{PUR}	CTRL 5		[<8>]	C= 8	
{SU}	SHIFT ↑CRSR↓		{GRN}	CTRL 6		{F1}	f1	
{GIU'}	↑CRSR↓		{BLU}	CTRL 7		{F2}	f2	
{SIN}	SHIFT ←CRSR→		{YEL}	CTRL 8		{F3}	f3	
{DES}	←CRSR→		[<1>]	C= 1		{F4}	f4	
{RVS}	CTRL 9		[<2>]	C= 2		{F5}	f5	
{OFF}	CTRL 0		[<3>]	C= 3		{F6}	f6	
{BLK}	CTRL 1		[<4>]	C= 4		{F7}	f7	
{WHT}	CTRL 2		[<5>]	C= 5		{F8}	f8	
{RED}	CTRL 3		[<6>]	C= 6				



OPEN

I sistemi di protezione a parola chiave per i programmi BASIC

di **S. K. Smith**
trad. ed adatt.
di **S. Colombo**

I servizi di telecomunicazione, le banche e in generale tutti quei servizi di informatica accessibili al pubblico che possono contenere informazioni riservate, richiedono all'utente una parola chiave per poter accedere al servizio.

Anche il programmatore BASIC può desiderare di proteggere il proprio database con un sistema a parola chiave, in modo da non permettere la lettura dei propri dati a persone non autorizzate.

La protezione a parola chiave usa sistemi molto diversi da quelli abitualmente impiegati per protezioni contro la duplicazione abusiva dei programmi: l'aggiunta di routine di controllo di parola chiave non interferisce in alcun modo con il LOAD o il SAVE di un programma. Altri utenti potranno quindi tranquillamente copiare il programma protetto da parola chiave, anche se, non conoscendola, non saranno in grado di farlo girare correttamente (a meno che, naturalmente, si tratti di programmatori smaliziati, in grado di eliminare la protezione).



Una semplice parola chiave

Il sistema più ovvio è talmente semplice da risultare utilizzabile solo come protezione dell'uso del programma da parte di bambini molto piccoli, o da chi non abbia la benché minima conoscenza della programmazione in BASIC. Supponendo di aver scelto "PIPPO" come parola chiave, sarà sufficiente aggiungere al programma le seguenti linee:

```
10 INPUT"PAROLA CHIAVE";A$
20 IFA$<>"PIPPO"THENPRINT"PAROLA
   RIFIUTATA":GOTO10
```

Chiamiamola routine di "livello 1": l'utente inserisce una variabile di stringa A$, e il programma la confronta con la stringa "PIPPO". Se il risultato del confronto è negativo, il programma ritorna alla linea 10, richiedendo nuovamente la parola chiave. Naturalmente, è possibile sostituire l'istruzione GOTO10 con un NEW per cancellare il programma dalla memoria nel caso la parola chiave sia errata.

Il problema più ovvio è che se qualcuno lista il programma, è perfettamente in grado di leggere la parola chiave in linea 20. Occorre quindi inibire il listing per nascondere la parola chiave.

Inibire il listing

Alcuni sistemi di protezione contro il listing di un programma sono già stati analizzati sul numero di Marzo di questa

rivista. Vediamone qui qualcun altro.
L'inserimento di caratteri shiftati in una istruzione REM può dare strani risultati: per esempio, REM [SHIFT+F] verrà listato come REM ASC. Molti caratteri shiftati vengono infatti interpretati dal computer, con il LIST, come parola chiave del BASIC.
Uno di questi caratteri shiftati ha un effetto particolarmente curioso: SHIFT+L dopo una REM fa fermare il listing di un programma con la stampa del messaggio "?SYNTAX ERROR". Aggiungete questa linea al programmino visto precedentemente:

```
15 REM L
```

La L sottolineata, con l'abituale codifica usata dalla nostra rivista, equivale a SHIFT+L.
Se vi trovate in modo maiuscolo/minuscolo, la L shiftata apparirà come una L maiuscola, mentre se siete in modo maiuscolo/grafico, la L shiftata genererà il carattere grafico serigrafato sulla parte anteriore destra del tasto L. Digitate LIST, e otterrete la visualizzazione delle sole linee 10 e 15. Il listing si interrompe con un "?SYNTAX ERROR". ("En passant", se usate SHIFT+L come codice ID di un disco, avrete la sua directory non listabile).
A questo punto un eventuale "sprotettore" appena un po' furbo può fare due cose: dal momento che la linea 15 è marcatamente la causa del problema, può semplicemente digitare LIST 16 –: ecco lì la parola chiave ben in vista! Oppure può semplicemente cancellare dal programma la linea 15 e tutto ridiventa perfettamente listabile.
Un'idea per confondere le idee a questo "sprotettore" in erba potrebbe consistere nel fare in modo di cambiare con qualche artificio il numero visualizzato di linea della REM, tale che essa, pur rimanendo per il computer la linea 15, appaia nel listato con un altro numero di linea.

Un falso numero di linea

Andate con il cursore su una riga vuota, premete il tasto SHIFT e contemporaneamente premete il tasto INST/DEL. Il tasto INSERT che avete premuto (cioè SHIFT+INST/DEL) vi pone in modo "insert", che è praticamente identico nel comportamento al modo virgolette, con una sola differenza: nel modo virgolette la pressione del tasto DEL provoca la cancellazione del carattere a sinistra del cursore, mentre in modo insert la pressione di questo tasto provoca l'apparizione di una T in campo inverso. Ecco come inserire dei caratteri di DEL in un'istruzione REM:

1. Digitate 15 REM"" (si noti che il tasto con le virgolette va premuto due volte).
2. Premete una volta il tasto INST/DEL per cancellare le ultime virgolette.
3. Premete CTRL contemporaneamente al tasto 9 (operazione coincidente con RVS ON), e quindi premete sette volte il tasto T.
4. Digitate, a seguire, 30 REM", quindi premete SHIFT ed L contemporaneamente.

Le T in campo inverso agiscono come caratteri DEL e cancellano i primi sette caratteri durante il LIST. Viene quindi stampata la scritta 30 REM.
Dando il LIST al programma si potrà quindi pensare che sia la linea 30 a seguire la linea 10, mentre in realtà si tratta della linea 15. Un utente non autorizzato che cerchi di listare con LIST30–, o cerchi di cancellare la linea 30, non potrà leggere la parola chiave contenuta in linea 20.
Passiamo ora a un diverso tipo di programma con parola chiave.

Giocare con i link di linea

Le linee BASIC sono contenute nella memoria del computer in base a regole ben definite. Un certo numero di PEEK potrà rivelarci quali siano queste regole.
Per prima cosa digitate NEW ed inserite questa linea: 10 REMAB. Ora digitate, senza numero di linea, quanto segue:

```
B=PEEK(43)+256*PEEK(44):FORI=BTO
B+9:PRINTI,PEEK(I):NEXT
```

Le locazioni 43 e 44 sono i puntatori alla prima locazione di memoria usata dal BASIC: 43 contiene il suo byte basso, e 44 il suo byte alto. Dopo aver premuto RETURN vedrete sullo schermo i seguenti numeri:

2049	9	byte basso del link
2050	8	byte alto del link
2051	10	byte basso numero linea
2052	0	byte alto numero linea
2053	143	codice token della REM
2054	65	valore ASCII di A
2055	66	valore ASCII di B
2056	0	byte di fine linea
2057	0	due zeri di fine program-
2058	0	ma

Un numero in formato byte basso/byte alto può essere letto per esteso con la formuletta:

PRINT (byte basso) + (byte alto)*256

Il link punta all'inizio della linea BASIC successiva. Poiché il byte basso del link è 9, e quello alto è 8, l'indirizzo di link sarà 9+8*256=2057. Se ci fosse un'altra linea BASIC in memoria, questa conterrebbe quindi il suo indirizzo di link nelle locazioni 2057 e 2058.
I link di linea funzionano puntandosi a vicenda: il primo punta al secondo, il secondo al terzo e così via. Si troverà inoltre sempre uno zero tra due linee, proprio nella locazione precedente il byte basso dell'indirizzo di link di una linea. L'ultima linea del programma punta al primo di una coppia di zeri, in modo che tre zeri consecutivi indicano la fine di un programma.
L'inserire degli zeri nel primo link di linea impedirà agli utenti non autorizzati del programma sia il LIST che il RUN del programma stesso, a meno che vengano digitate opportune POKE, che in questo caso assumono la funzione di parola chiave.
Per passare alla pratica provate a caricare in memoria un programma che desiderate proteggere in questa maniera, quindi seguite questa procedura:

1. Listate il programma per essere sicuri che si trovi in memoria.
2. Digitate B=PEEK (43) + 256*PEEK (44) : PRINTB. Annotatevi il numero che verrà visualizzato.
3. Digitate PRINTPEEK (B), PEEK (B+1) ed annotatevi i due numeri visualizzati.
4. Digitate POKEB,0 : POKEB+1,0. Quindi date il LIST.

Il programma apparentemente se n'è andato. Potete comunque salvarlo su nastro o disco, perché è naturalmente ancora in memoria, anche se invisibile.
Saranno sufficienti due POKE per riottenere il programma listabile ed eseguibile. Se chiamiamo rispettivamente A il primo numero che avete annotato, B il secondo e C il terzo, potrete digitare:

```
POKEA,B:POKEA+1,C
```

e tutto ritornerà normale.

Una parola chiave numerica

Ci sono un paio di variazioni per perfezionare questa tecnica: digitate NEW, seguito da questo programma:

```
10 PRINT"PAROLA CHIAVE":INPUTS,L
   ,H:POKES,L:POKES+1,H
20 PRINT"CIAO"
```

Listatelo, per assicurarvi che sia effettivamente in memoria. Quindi digitate la seguente linea in modo diretto:

```
B=PEEK(43)+256*PEEK(44):FORJ=B+5
TOB+80:PRINT-J*(PEEK(J-1)=0);:NE
XT
```

Lo schermo si riempirà con molti zeri frammisti ad altri numeri. Annotatevi il primo numero diverso da zero e chiamiamolo locazione chiave. Poniamo che esso sia 2090 (se fosse un numero diverso, usate quello). Ciò significa che il link alla linea successiva (linea 20) si trova nelle locazioni 2090–2091.
Ora digitate PRINTPEEK (2090), PEEK (2091) ed annotatevi i due numeri visualizzati.
Poniamo che essi siano 57 e 8.
Date ora POKE2090,0:POKE2091,0 (ricordate di non usare questi numeri, se la vostra locazione chiave è diversa).
Il porre a zero gli indirizzi di link rende il listing impossibile dopo la linea 10: provate a dare il LIST! Se conoscete la parola chiave di tre numeri, potrete in-

vece avere accesso all'intero programma.
Nell'esempio fatto date il RUN e digitate come risposta all'INPUT 2090,57,8. Tutto ritornerà normale.

Un programma infinito

Ricordate che le locazioni 43 e 44 puntano al primo link di linea, ed il primo link di linea punta al secondo. Che succederà se facciamo in modo che il primo link di linea punti a se stesso? Proviamo. Digitate NEW ed inserite questo breve programma:

```
10 REM PRIMA LINEA
20 REM SECONDA LINEA
```

Digitate quindi in modo diretto:

```
B=PEEK(43)+256*PEEK(44):PRINTB,P
EEK(B),PEEK(B+1)
```

Annotatevi i tre numeri e digitate:

```
POKEB,PEEK(43):POKEB+1,PEEK(44)
```

Ora la prima linea punta su se stessa. Date il LIST e vedrete visualizzata all'infinito la linea 10. Infatti, dopo aver listato la prima linea il computer cerca in memoria e legge il link alla seconda linea. Dal momento che abbiamo fatto in modo che il link punti su se stesso, la prima linea verrà listata all'infinito. Per riottenere il programma corretto ripristinate il link originario, "pokando" i numeri che vi siete annotati.
Come visto all'inizio dell'articolo, una REM seguita da particolari caratteri può rendere una linea invisibile a chiunque cerchi di listarla. Se si usa questo sistema per rendere invisibile la prima linea, e quindi le si modifica l'indirizzo di link per fare in modo che punti su se stessa, il comando LIST sembrerà disabilitato. Il computer stamperà infatti una serie infinita di linee invisibili.
Una combinazione delle tecniche esaminate può essere in grado di proteggere i vostri programmi da un buon numero di occhi troppo curiosi. Tenete conto tuttavia che ad ogni tecnica di protezione corrisponde una tecnica di sprotezione altrettanto sofisticata, e che le migliori tecniche di protezione ricorrono totalmente al linguaggio macchina e possono addirittura essere ancora più complicate dello stesso programma da proteggere.

Un'occhiata allo SMAU '85

di **M. Cristuib Grizzi**

Dal 19 al 24 Settembre si è tenuta a Milano, come di consueto, la rassegna SMAU 1985: in esposizione una gran varietà di personal computer, periferiche ed accessori vari, sia per l'hobby che per l'"office automation".

Malgrado la tanto discussa flessione nel mercato statunitense dei personal, a giudicare dalle presenze e dall'afflusso di visitatori a questa edizione dello SMAU, si direbbe che da noi il fenomeno non esista quasi. In realtà a casa nostra il personal computer è ancora da considerarsi come fenomeno nascente, e la sua diffusione negli uffici deve ancora diventare massiccia, forse perché restano ancora da vincere resistenze psicologiche ed abitudini più che radicate da parte dei potenizali utenti.
Non si pone quindi — a differenza di quanto avviene oltre Atlantico — il problema della saturazione del mercato e della forzata riconversione od addirittura chiusura di molte aziende del settore elettronico-informatico. Buon per noi, e buon per tutte quelle aziende del settore che hanno trovato nello SMAU un'ottima vetrina di presentazione per propri prodotti più tecnologicamente avanzati.
Tuttavia, malgrado la presenza di questo gran numero di aziende produttrici sia famose che meno famose, quasi nulla tra il parco macchina presentato si può classificare come novità assoluta nel settore dell'home computer.
Ai nostri lettori interesserà comunque sapere della presenza allo stand Commodore del nuovo C128 — a proposito del quale abbiamo già parlato piuttosto diffusamente su queste pagine — accompagnato dal suo drive dedicato 1570 e dal nuovo monitor a colori 1901.
Ancora nulla invece per quello che dovrebbe essere il cavallo di battaglia della Commodore nei prossimi anni: l'"Amiga", un computer del quale avremo occasione di parlare piuttosto diffusamente sul prossimo numero, e del quale anticipiamo soltanto che è un "Macintosh style" che supera di gran lunga il "maestro" in prezzo e prestazioni (risoluzione grafica eccezionale, menu a scorrimento, finestratura integrale, multitasking, due canali audio e stereo, mouse, animazioni, e tante altre leccornie...)



Una veduta dello Stand Commodore con il nuovo C128



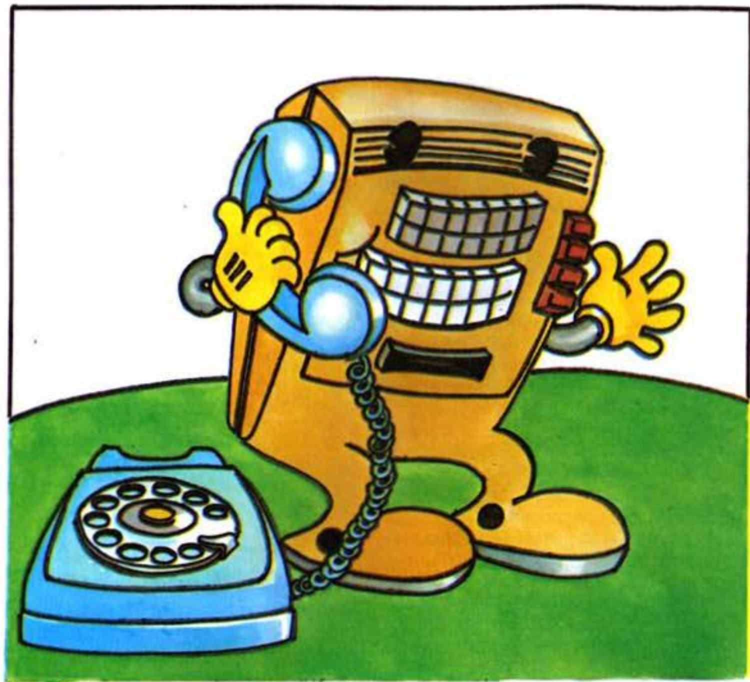
Ecco come appariva lo Stand Sinclair

Presenti anche i PC Commodore, dei compatibili PC–IBM che si collocano in una fascia di mercato piuttosto ricca di agguerrita concorrenza, e destinata comunque quasi totalmente al settore gestionale. Sparsi qui e là, i soliti C64, C16 e Plus/4 con i giochini più classici, presi d'assalto da nugoli di visitatori al di sotto dei vent'anni.
La Sinclair presentava come unica novità un drive per microfloppy da tre pollici e mezzo per il QL, che è stato italianizzato nei messaggi del sistema operativo.
È da registrare invece la prima comparsa massiccia ad una rassegna del genere di strumenti e servizi per telecomunicazione tra computer: finalmente anche da noi si sta facendo più sentita l'esigenza della rapida acquisizione e scambio di dati tra aziende e tra singoli utenti di personal computer, parallelamente ad un nascente interesse per le telecomunicazioni da parte di molti hobbisti. Tra le prime iniziative italiane di banca dati aperta a tutti gli utenti di personal munito di modem, le Pagine Gialle Elettroniche sono un servizio offerto dalla Seat che, con un canone di abbonamento di circa centomila lire annue, offre tutte le informazioni contenute nelle Pagine Gialle di tutta Italia, con il vantaggio della ricerca per chiavi multiple dell'informazione desiderata. Gran parte delle informazioni contenute nel database sono inoltre integrate da note aggiuntive (per esempio capitale sociale, numero dipendenti, partita IVA, etc.).
Come già lo scorso anno, presenti anche i due stand della J.soft in cui venivano presentati i nostri libri, riviste, e soprattutto i pacchetti software professionali per i più diffusi personal.
In conclusione, una rassegna che, pur non presentando strepitose novità, ha confermato una volta di più l'interesse di privati ed aziende per il personal computer.

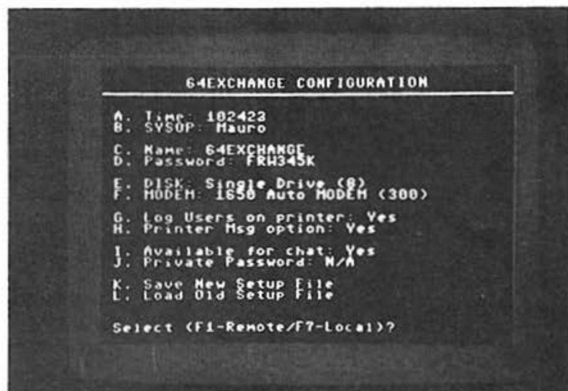
Telecomunicazioni: primi passi

di **S. Bateman**
trad. ed adatt.
di **S. Colombo**

Entrare con il proprio computer nel mondo delle telecomunicazioni può essere una cosa estremamente eccitante e divertente sotto molti aspetti. Tuttavia, per fare i primi esperimenti con un modem occorre avere almeno un'infarinatura circa le problematiche connesse con le telecomunicazioni tra computer.



Recentemente gli scrittori Stephen King e Peter Straub hanno scritto "a quattro mani" un nuovo libro del genere horror: malgrado essi risiedessero in città americane lontanissime tra loro, durante la stesura del libro si scambiavano quotidianamente ed istantaneamente le porzioni di testo che man mano venivano scritte. King, dalla sua casa nel Maine, riceveva regolarmente nuovo materiale e revisioni via telefono – direttamente nel suo word processor – da parte di Straub in California.
King e Straub non sono i primi ad usare questo tipo di telecomunicazioni: il loro è solo un esempio del modo in cui le possibilità di comunicazioni tra computer stanno cambiando il nostro modo di lavorare e – perché no – di divertirci.
Negli Stati Uniti migliaia di utenti Commodore già fanno uso regolare delle possibilità di comunicazione dei loro computer attraverso le linee telefoniche e, se le previsioni effettuate dalle case produttrici sono corrette, centinaia di persone ogni mese si aggiungono a questa schiera.
Per molti utenti di computer, tuttavia, la telecomunicazione è una pratica creduta frustrante e complicata. Anche chi ha già una notevole pratica di programmazione o nell'uso di programmi utili come i word processor, e sia psicologicamente attirato dalle telecomunicazioni, al solo pensiero di avere a che fare con problemi di parità, bit di stop, lunghezza della parola, ecc., non tenta nemmeno una pur timida sperimentazione.
In realtà occorre semplicemente imparare un certo numero di nozioni fondamentali, dopo di che ci si potrà avvicinare al vastissimo mondo delle telecomunicazioni per trarne tutte le soddisfazioni che esso è in grado di offrire.
Il modo più semplice per capire i problemi connessi alle telecomunicazioni è seguire la strada percorsa da una singola informazione, dal momento in cui la digitate sulla tastiera del vostro Commodore fino al momento in cui raggiunge le linee telefoniche.
Inviare un messaggio attraverso il computer significa seguire certe regole fisse, proprio come una vacanza richiede necessariamente un piano di viaggio, un veicolo e una destinazione. Un computer che effettui delle telecomunicazioni richiede analoghe definizioni di alcuni parametri.
Innanzi tutto vediamo quali sono i com-



ponenti necessari per effettuare delle telecomunicazioni: come molti utenti di computer hanno ormai imparato, occorrono fondamentalmente tre cose. Un *modem*, che collega il computer a un *telefono*, e un *programma di comunicazione*, che vi permetta di far sapere al computer e al modem che cosa, quando, dove e come volete comunicare.

Per i computer Commodore esiste già una larga gamma di modem e di programmi di telecomunicazione, le cui prestazioni e complessità sono direttamente proporzionali al loro prezzo.

Prima di esaminare come questi due componenti – modem e programma di comunicazione – lavorano insieme, ritorniamo ancora un momento alla tastiera del computer ed al nostro telefono di casa.

L'apparecchio telefonico è un apparato *analogico*: trasmette suoni continui oscillanti tra due valori, minimo e massimo, proprio come un pittore che, disponendo solo del bianco e del nero, potrebbe mischiarli per ottenere un'infinita varietà di grigi. Il computer, al contrario, rappresenta ogni carattere in forma *digitale*: ogni bit della memoria contiene uno 0 oppure un 1, e il pittore del paragone precedente può dipingere solo usando il nero oppure il bianco, il grigio non è ammesso.

Come ormai tutti dovrebbero sapere, la codifica dei caratteri all'interno del computer si rifà alle norme ASCII (American Standard Code for Information Interchange). In particolare i computer Commodore usano una propria forma di ASCII, spesso chiamata Commodore ASCII o PET ASCII, che differisce leggermente dallo standard. Una delle differenze è che il vero ASCII usa sette bit per carattere, mentre il Commodore ASCII ne usa otto. Se si desidera comunicare con un altro computer (che non necessariamente dovrà essere un altro Commodore), i caratteri che digitiamo sulla tastiera dovranno per prima cosa essere convertiti in vero ASCII. Ecco una delle prime funzioni svolte dall'insieme modem/programma di comunicazione: quest'ultimo converte i caratteri digitati in vero ASCII e li invia al modem, il quale provvede a sua volta a convertirli in segnali sonori, che possono essere trasmessi attraverso la linea telefonica.

I modem più popolari per computer Commodore sono attualmente il 1600 VIC MODEM e il 1650 AutoModem – prodotti dalla stessa Commodore –, che si adattano a VIC 20, C64 e Plus/4 (il C16 non è dotato della possibilità di collegamento a un modem). Esiste poi una vasta gamma di modem prodotti da altre aziende, perfettamente compatibili con i computer Commodore. Anche se le capacità di un modem variano a seconda del suo prezzo di acquisto, la funzione fondamentale di ogni modem è MODulare il segnale digitale proveniente dal computer in forma analogica e DEModulare il segnale proveniente in forma analogica dal telefono, trasformandolo in forma digitale.

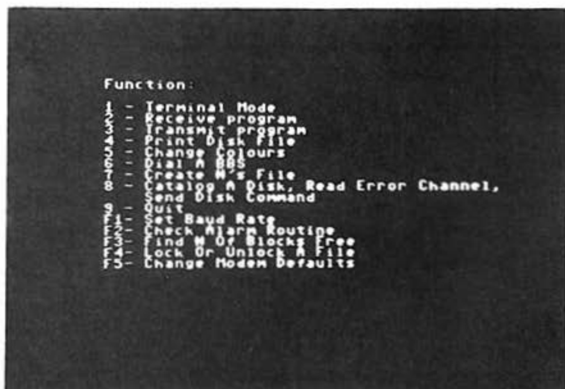
Ci sono tre scelte da operare per regolare il modem e tre altre scelte per regolare invece il programma di comunicazione. Una volta chiariti i contenuti di queste scelte si potrà tranquillamente iniziare la pratica delle telecomunicazioni.

Le tre cose che il modem deve "sapere" prima di ricevere qualunque messaggio dal computer sono:

1. Se si deve predisporre a ricevere o ad inviare dei dati.
2. A quale velocità viaggiano i dati.
3. Se i messaggi inviati devono o meno essere dotati di una eco con funzioni di controllo degli errori.

L'eco consiste semplicemente nel reinviare al computer trasmittente i caratteri ricevuti, in modo che esso possa confrontarli con quelli inviati e rilevare eventuali discrepanze dovute a disturbi sulla linea telefonica.

Usando i termini di "gergo", si ha che il punto 1 corrisponde al *mode* (originate o Answer), il punto 2 al *baud rate* (velocità misurata in bit al secondo) e il punto 3 corrisponde al *duplex* (half duplex = senza eco, full duplex = con eco). Se desideriamo quindi inviare un messaggio, regoleremo il modem su *originate*, mentre il computer che dovrà ricevere sarà regolato su *answer*. La velocità di trasmissione più comune disponibile con i modem economici è di 300 baud (che corrisponde al trasferimento di circa 30 caratteri al secondo). Modem più sofisticati possono arrivare anche a 2400 baud, ma con le linee telefoniche che ci ritroviamo ad avere una velocità bassa risulta senz'altro molto più affidabile, pur richiedendo un tempo di impegno superiore della linea telefonica. Entrambi i modem – trasmittente e ricevente – dovranno essere regolati sulla



stessa velocità.
La terza opzione, il duplex, come detto consente di scegliere tra la possibilità di eco dei caratteri e la trasmissione normale. Con la trasmissione in eco (*full duplex*), digitando una frase da trasmettere, ce la vedremo duplicata sul video dopo pochi secondi (cioè dopo il tempo necessario ad inviare i caratteri all'altro computer ed a compiere il percorso inverso).
Il full duplex è ovviamente più lento dell'half duplex, ed è di uso conveniente solo se si trasmettono dati veramente importanti, in cui un solo numero trasmesso male provocherebbe molti inconvenienti (si pensi ad esempio alla trasmissione di un programma con molte linee DATA). Poiché molto spesso sia il modem che il programma di comunicazione hanno la possibilità di regolazione tra half e full duplex, occorre che entrambi siano regolati sulla stessa opzione. Se vi capitasse ad esempio di digitare in trasmissione la stringa "PIPPO", e di vedere sul video un eco "PPIIPPOO", probabilmente il vostro modem e il programma di comunicazione non sono regolati in accordo.
I programmi di comunicazione variano enormemente le loro potenzialità da caso a caso: ci sono semplici programmi in BASIC e complessi e lunghi programmi in linguaggio macchina. Da caso a caso cambia, ovviamente, la velocità di esecuzione ed il numero di opzioni disponibili. Tra queste ultime ce ne sono tre fondamentali: bit di stop, lunghezza della parola e parità. Ritornando all'inglese (usato più che mai nel gergo delle telecomunicazioni), si ha che l'opzione di *stop bit* è un'opzione si/no. Il bit di stop viene aggiunto automaticamente alla fine di ogni stringa di sette bit, per segnalare la fine di un singolo carattere.
L'opzione di *word lenght* stabilisce se il computer deve considerare i caratteri composti da sette o da otto bit, mentre l'opzione di *parity*, in due parole, serve a fornire al computer un ulteriore sistema di controllo degli errori in trasmissione, settando un ulteriore bit.
Sembra complicato, ma quasi tutti i buoni programmi di telecomunicazione prevedono per default i valori comunemente più usati (full duplex, 300 baud, originate, no parity, 1 stop bit, word lenght 8).
Esiste poi una serie di opzioni che può essere o meno presente nei programmi di telecomunicazione, a seconda della loro qualità. Diamo ora un piccolo glossario dei termini più usati nelle telecomunicazioni:

Auto answer: la possibilità da parte del sistema computer/modem di rispondere automaticamente a chiamate telefoniche provenienti da altri computer.

Auto dial: la possibilità da parte del sistema computer/modem di effettuare automaticamente chiamate a dati numeri telefonici e di riprovare nel caso la linea sia occupata.

Buffer: un'area nella memoria del computer usata per l'immagazzinamento temporaneo di dati durante la trasmissione o la ricezione.

Bulletin Board System: un servizio pubblico basato su computer per lo scambio di programmi e informazioni (prossimamente, ci auguriamo, anche in Italia).
Carrier signal: il segnale con cui un computer indica a un altro computer che è pronto a stabilire la comunicazione.
Downloading: il ricevere dati, come un file dati o un programma, da un altro computer.
Uploading: l'inviare dati, come un file dati o un programma, a un altro computer.

MLX

di **C. Brannon**
trad. e adatt.
di **M. Cristuib Grizzi**

MLX è un programma che permette di inserire listati in linguaggio macchina esenti in modo assoluto da errori e senza la perdita di tempo del dover battere e controllare lunghe sequenze di istruzioni DATA.
Molti dei nostri listati di programmi in linguaggio macchina hanno il formato MLX (li riconoscete dal fatto che sono esclusivamente numerici) e richiedono quindi che MLX sia caricato in memoria ed eseguito prima della battitura del listato.
MLX vi chiederà l'indirizzo della locazione di partenza e quello della locazione finale del programma da caricare. Questi valori sono sempre indicati nell'articolo che accompagna il listato. MLX vi segnala automaticamente gli errori di battitura MENTRE STATE DIGITANDO IL LISTATO e vi chiede di reinserire la linea errata. L'ultimo numero battuto di ogni linea rappresenta il checksum e viene visualizzato in reverse.
Sono inoltre disponibili altri comandi, quali:
SHIFT-N nuovo indirizzo: permette di cambiare l'indirizzo della linea che volete battere ed è utile nel caso si inseriscano i listati in più parti.
SHIFT-D display: lista i dati inseriti tra due indirizzi di inizio e fine.
SHIFT-L load: carica un file da nastro o SHIFT-S save: salva su nastro o disco un file in formato MLX.
Una volta battuto il listato, e salvatolo tramite MLX, si potrà caricare direttamente il programma con un'istruzione LOAD "nome del file",1,1 per il registratore, oppure LOAD "nome del file",8,1 per l'unità a dischi, salvo diverse indicazioni date negli articoli corrispondenti ai vari programmi. Il programma sarà generalmente mandato in esecuzione con una SYS (indirizzo di partenza).
La versione originaria di MLX per C64 ha subito, dalla prima pubblicazione, diversi ritocchi e miglioramenti, fino alla versione 2.0 qui presentata.
È stato in primo luogo modificato il colore di fondo dello schermo e del bordo, in modo da risultare meno stancante alla vista; sono state ulteriormente perfezionate le routine in linguaggio macchina di salvataggio e caricamento dei programmi; è cambiata la forma del cursore e, cosa più importante, è stato aggiunto un tastierino numerico per gli utenti che si trovano più a loro agio con una diversa disposizione dei tasti.
Oltre a poter usare i soliti tasti numerici, sono stati ridefiniti alcuni tasti alfabetici, in modo da generare ugualmente dei numeri. I tasti ridefiniti sono i seguenti:

```
UIO                    7 8 9
HJKL       diventano 0 4 5 6
M,.                    1 2 3
```

Le persone abituate ad usare tali tastierini accresceranno senz'altro, in brevissimo tempo, la velocità di battitura dei programmi in linguaggio macchina.

MLX

```
100 PRINT"{CLR}{CYN}";CHR$(142);CHR$(8);:
    POKE53280,0:POKE53281,0         :rem  71
101 POKE788,52:REM DISABILITA RUN/STOP
                                    :rem  89
110 PRINT"{RVS}{ 40 SPAZI}";        :rem 176
120 PRINT"{RVS}{ 15 SPAZI}{DES}{OFF}[<*>]
    £{RVS}{DES} {DES}{ 2 SPAZI}[<*>]{OFF}
    T[<*>]£{RVS}£{RVS}{ 13 SPAZI}";
                                    :rem 250
130 PRINT"{RVS}{ 15 SPAZI}{DES}[<N>][<H>]
    {DES} { 2 DES} {OFF}£{RVS}£[<*>]{OFF}
    [<*>]{RVS}{ 13 SPAZI}";         :rem 220
140 PRINT"{RVS}{ 40 SPAZI}"         :rem 120
200 PRINT"{ 2 GIU'}{PUR}EDITOR DI CODICE
    MACCHINA VERSIONE 2.0{ 3 GIU'}"
                                    :rem 126
210 PRINT"[<5>]{ 2 SU} INDIRIZZO DI PARTE
    NZA{ 2 SPAZI}";:INPUTS:F=1-F:C$=CHR$(
    31+119*F)                       :rem  85
220 IFS<256OR(S>40960ANDS<49152)ORS>53247
    THENGOSUB3000:GOTO210           :rem 235
225 PRINT:PRINT:PRINT               :rem 180
230 PRINT"[<5>]{ 2 SU} INDIRIZZO CONCLUSI
    VO{ 3 SPAZI}";:INPUTE:F=1-F:C$=CHR$(3
    1+119*F)                        :rem  92
240 IFE<256OR(E>40960ANDE<49152)ORE>53247
    THENGOSUB3000:GOTO230           :rem 183
250 IFE<STHENPRINTC$;"{RVS}INDIRIZZO CONC
    LUSIVO<INDIRIZZO INIZIALE"      :rem 200
255 IFE<STHENGOSUB1000:GOTO230      :rem 119
260 PRINT:PRINT:PRINT               :rem 179
300 PRINT"{CLR}";CHR$(14):AD=S:POKEV+21,0
                                    :rem 225
310 A=1:PRINTRIGHT$("0000"+MID$(STR$(AD),
    2),5);":";:FORJ=ATO6            :rem 227
315 FORJ=ATO6                       :rem  33
320 GOSUB570:IFN=-1THENJ=J+N:GOTO320
                                    :rem 228
390 IFN=-211THEN710                 :rem  62
400 IFN=-204THEN790                 :rem  64
410 IFN=-206THENPRINT:INPUT"{GIU'} INSERI
    RE IL NUOVO INDIRIZZO";ZZ       :rem  13
414 IFN=-206THENIFZZ<SORZZ>ETHENPRINT"
    {RVS}ESCE DAL CAMPO DI VALORI INDICAT
    O"                              :rem 105
415 IFN=-206THENIFZZ<SORZZ>ETHENGOSUB1000
    :GOTO410                        :rem 202
417 IFN=-206THENAD=ZZ:PRINT:GOTO310
                                    :rem 238
420 IFN<>-196THEN480                :rem 133
430 PRINT:INPUT"LISTATO:DA";F:PRINT"
    { 9 SPAZI}A";:INPUTT            :rem 183
440 IFF<SORF>EORT<SORT>ETHENPRINT"MINIMO"
    ;S;" MASSIMO";E;"![<5>]":GOTO430
                                    :rem 147
450 FORI=FTOTSTEP6:PRINT:PRINTRIGHT$("000
```

```
   0"+MID$(STR$(I),2),5);":";   :rem   30
451 FORK=0TO5:N=PEEK(I+K):PRINTRIGHT$("00
   "+MID$(STR$(N),2),3);",";     :rem   66
460 GETA$:IFA$>""THENPRINT:PRINT:GOTO310
                                 :rem   25
470 NEXTK:PRINTCHR$(20);:NEXTI:PRINT:PRIN
   T:GOTO310                     :rem   50
480 IFN<0THENPRINT:GOTO310       :rem  168
490 A(J)=N:NEXTJ                 :rem  199
500 CKSUM=AD-INT(AD/256)*256:FORI=1TO6:CK
   SUM=(CKSUM+A(I))AND255:NEXT  :rem  200
510 PRINTCHR$(18);:GOSUB570:PRINTCHR$(146
   );                            :rem   94
511 IFN=-1THENA=6:GOTO315        :rem  254
515 PRINTCHR$(20):IFN=CKSUMTHEN530
                                 :rem  122
520 PRINT:PRINT"{RED}LA LINEA E' STATA IN
   SERITA IN MANIERA"            :rem  157
525 PRINT"ERRATA. RIPETERE[<5>]":PRINT:GO
   SUB1000:GOTO310               :rem   27
530 GOSUB2000                    :rem  218
540 FORI=1TO6:POKEAD+I-1,A(I):NEXT:POKE54
   272,0:POKE54273,0             :rem  227
550 AD=AD+6:IFAD<ETHEN310        :rem  212
560 GOTO710                      :rem  108
570 N=0:Z=0                      :rem   88
580 PRINT"[<£>]";                :rem   81
581 GETA$:IFA$=""THEN581         :rem   95
582 AV=-(A$="M")-2*(A$=",")-3*(A$=".")-4*
   (A$="J")-5*(A$="K")-6*(A$="L")
                                 :rem   41
583 AV=AV-7*(A$="U")-8*(A$="I")-9*(A$="O"
   ):IFA$="H"THENA$="0"          :rem  134
584 IFAV>0THENA$=CHR$(48+AV)     :rem  134
585 PRINTCHR$(20);:A=ASC(A$):IFA=13ORA=44
   ORA=32THEN670                 :rem  229
590 IFA>128THENN=-A:RETURN       :rem  137
600 IFA<>20THEN630               :rem   10
610 GOSUB690:IFI=1ANDT=44THENN=-1:PRINT"
   {SIN} {SIN}";:GOTO690         :rem  172
620 GOTO570                      :rem  109
630 IFA<48ORA>57THEN580          :rem  105
640 PRINTA$;:N=N*10+A-48         :rem  106
650 IFN>255THENA=20:GOSUB1000:GOTO600
                                 :rem  229
660 Z=Z+1:IFZ<3THEN580           :rem   71
670 IFZ=0THENGOSUB1000:GOTO570   :rem  114
680 PRINT",";:RETURN             :rem  240
690 S%=PEEK(209)+256*PEEK(210)+PEEK(211)
                                 :rem  149
691 FORI=1TO3:T=PEEK(S%-I)       :rem   67
695 IFT<>44ANDT<>58THENPOKES%-I,32:NEXT
                                 :rem  205
700 PRINTLEFT$("{ 3 SIN}",I-1);:RETURN
                                 :rem    7
710 PRINT"{CLR}{RVS}*** SAVE ***
   { 3 GIU'}"                    :rem  236
715 PRINT"{ 2 GIU'}PREMERE {RVS}RETURN
   {OFF} PER USCIRE DAL SAVE{GIU'}"
                                 :rem  103
720 F$="":INPUT"{GIU'}NOME DEL FILE";F$:I
   FF$=""THENPRINT:PRINT:GOTO310
                                 :rem   42
730 PRINT:PRINT"{ 2 GIU'}{RVS}N{OFF}ASTRO
    O {RVS}D{OFF}ISCO: (N/D)"    :rem  128
740 GETA$:IFA$<>"N"ANDA$<>"D"THEN740
                                 :rem   30
750 DV=1-7*(A$="D"):IFDV=8THENF$="0:"+F$:
   OPEN15,8,15,"S"+F$:CLOSE15    :rem  212
760 T$=F$:ZK=PEEK(53)+256*PEEK(54)-LEN(T$
   ):POKE782,ZK/256              :rem    3
762 POKE781,ZK-PEEK(782)*256:POKE780,LEN(
   T$):SYS65469                  :rem  109
763 POKE780,1:POKE781,DV:POKE782,1:SYS654
   66                            :rem   69
765 K=S:POKE254,K/256:POKE253,K-PEEK(254)
   *256:POKE780,253              :rem   17
766 K=E+1:POKE782,K/256:POKE781,K-PEEK(78
   2)*256:SYS65496               :rem  235
770 IF(PEEK(783)AND1)OR(191ANDST)THEN780
                                 :rem  111
775 PRINT"{GIU'}FATTO.{GIU'}":GOTO310
                                 :rem  201
780 PRINT"{GIU'}ERRORE NEL SAVE-RIPROVA!"
   :IFDV=1THEN720                :rem  104
781 OPEN15,8,15:INPUT#15,E1$,E2$:PRINTE1$
   ;E2$:CLOSE15:GOTO720          :rem  103
790 PRINT"{CLR}{RVS}*** LOAD ***
   { 2 GIU'}"                    :rem  212
795 PRINT"{ 2 GIU'}PREMERE {RVS}RETURN
   {OFF} PER USCIRE DAL LOAD{GIU'}"
                                 :rem   96
800 F$="":INPUT"{ 2 GIU'}NOME DEL FILE";F
   $:IFF$=""THENPRINT:GOTO310    :rem  115
810 PRINT:PRINT"{ 2 GIU'}{RVS}N{OFF}ASTRO
    O {RVS}D{OFF}ISCO: (N/D)"    :rem  127
820 GETA$:IFA$<>"N"ANDA$<>"D"THEN820
                                 :rem   28
830 DV=1-7*(A$="D"):IFDV=8THENF$="0:"+F$
                                 :rem  157
840 T$=F$:ZK=PEEK(53)+256*PEEK(54)-LEN(T$
   ):POKE782,ZK/256              :rem    2
841 POKE781,ZK-PEEK(782)*256:POKE780,LEN(
   T$):SYS65469                  :rem  107
845 POKE780,1:POKE781,DV:POKE782,1:SYS654
   66                            :rem   70
850 POKE780,0:SYS65493           :rem   11
860 IF(PEEK(783)AND1)OR(191ANDST)THEN870
                                 :rem  111
865 PRINT"{GIU'}FATTO.":GOTO310  :rem  184
870 PRINT"{GIU'}ERRORE NEL LOAD-RIPETI!
   {GIU'}":IFDV=1THEN800         :rem   19
880 OPEN15,8,15:INPUT#15,E1$,E2$:PRINTE1$
   ;E2$:CLOSE15:GOTO800          :rem  102
1000 REM CICALINO                :rem  231
1001 POKE54296,15:POKE54277,45:POKE54278,
    165                          :rem  207
1002 POKE54276,33:POKE54273,6:POKE54272,5
                                 :rem   42
1003 FORT=1TO200:NEXT:POKE54276,32:POKE54
    273,0:POKE54272,0:RETURN     :rem  202
2000 REM CAMPANELLO              :rem  130
2001 POKE54296,15:POKE54277,0:POKE54278,2
    47                           :rem  152
2002 POKE54276,17:POKE54273,40:POKE54272,
    0                            :rem   86
2003 FORT=1TO100:NEXT:POKE54276,16:RETURN
                                 :rem   57
3000 PRINTC$;"{RVS} NON IN PAGINA ZERO O
    SU{DES}ROM ":GOTO1000        :rem  240
```

REM:HW

Hardware

di **J. J. Shaughnessy**
trad. ed adatt.
di **M. Cristuib Grizzi**

Come riallineare la testina del drive 1541

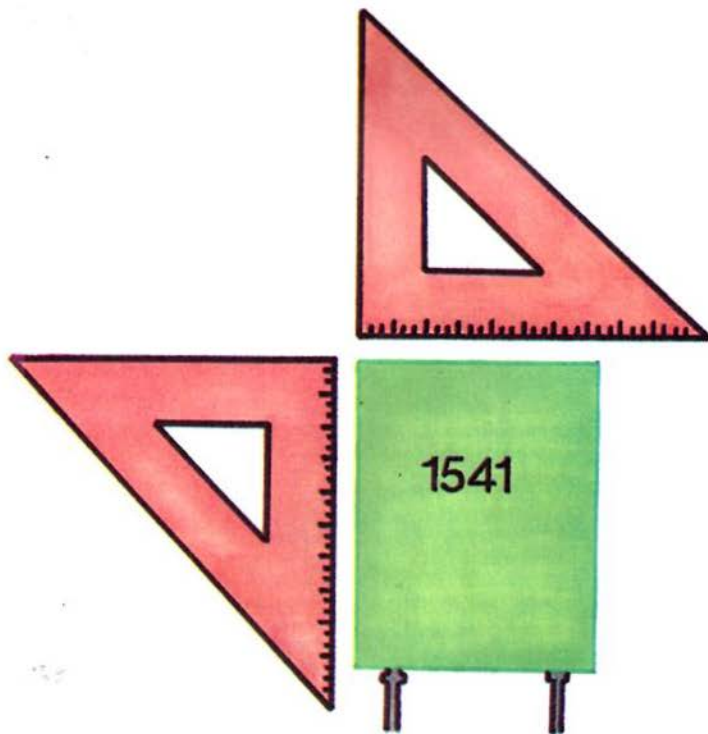
Proprio come il registratore a cassette, anche il disk drive va soggetto a disallineamenti della testina di lettura/scrittura che – alla lunga – impediscono la corretta lettura dei dischi precedentemente registrati. Ecco alcuni consigli per effettuare da soli il riallineamento della testina.

Questo articolo vi fornisce dei consigli tali da ottenere un risparmio, in spese di manutenzione del drive, sufficiente per abbonarsi a SUPERCOMMODORE per molti anni a venire... Tenendo conto che per l'operazione di riallineamento delle testine sono mediamente richieste dai tecnici dalle cento alle duecentomila lire, il risparmio ottenuto con il compiere da sé l'operazione risulta senz'altro notevole.

Le istruzioni date in questo articolo comportano operazioni su un apparecchio elettrico (il drive) sotto tensione: esiste quindi il pericolo di subire una scarica elettrica in caso di qualche errore operativo. Si sconsiglia perciò di effettuare le operazioni qui descritte a chi non sia perfettamente in grado di eseguirle correttamente. SUPERCOMMODORE non si assume pertanto alcuna responsabilità circa eventuali danni a persone o cose dovuti alle operazioni descritte in questo articolo.

Le operazioni da compiere per il riallineamento della testina invalidano inoltre la garanzia del drive, quindi prima di procedere assicuratevi che questa sia scaduta (in caso contrario, visto che l'operazione è coperta da garanzia, vi conviene farla effettuare da un tecnico).

L'unico attrezzo richiesto per l'operazione è un piccolo cacciavite a croce.



Informazioni preliminari

Per prima cosa vediamo come le informazioni (programmi, dati) sono contenute nei dischetti. Ogni disco possiede 35 tracce, disposte concentricamente, con la traccia numero 1 all'esterno e la 35 la più interna. Ogni traccia possiede da 17 a 21 blocchi ed ogni blocco contiene 256 byte di dati. La traccia centrale è la 18, con funzioni speciali.

Il primo blocco (blocco 0) della traccia 18 viene usato per immagazzinare il nome ed il codice ID che avete assegnato al disco, altri codici speciali e la BAM (Block Availability Map). Il drive usa la BAM per sapere quali blocchi risultano liberi per il salvataggio di nuovi dati e quali sono invece occupati da precedenti registrazioni. I blocchi da 1 a 18 della traccia 18 sono invece usati per contenere la directory del disco.

Quest'ultima contiene i nomi dei file con le informazioni circa il tipo di file, la sua lunghezza (numero di blocchi occupati) e il suo punto di inizio all'interno del disco. Tutti i blocchi rimanenti del disco servono a contenere i programmi o i dati (664 blocchi su un disco vuoto).
L'immagazzinamento dei file inizia sulle tracce 17 e 19 e procede in direzione radiale, sia verso il centro che verso l'esterno, a partire dalla traccia centrale, con le tracce 1 e 35 che rappresentano quindi le ultime tracce usabili prima che il disco sia completamente pieno.
Quando il drive comincia ad uscire dal corretto allineamento le prime tracce a diventare illeggibili sono queste ultime, la 1 e la 35. Noterete che è possibile caricare programmi nella parte iniziale della directory, mentre quelli nella sua parte finale si caricheranno con difficoltà (o non si caricheranno affatto).
In generale, i programmi elencati per primi nella directory sono immagazzinati nella parte centrale del disco, dal momento che sono stati salvati per primi. Tuttavia, quando il disallineamento della testina aumenta il drive non è nemmeno più in grado di caricare la directory.
Un altro segno del progressivo disallineamento può essere il dilatarsi del tempo di caricamento dei programmi, con l'accendersi e spegnersi della luce rossa del drive, e l'insorgere di strani rumori dovuti al resettarsi della testina contro il punto di stop quando il tentativo di leggere un blocco ha esito negativo.
Su un drive perfettamente allineato la luce rossa rimane completamente accesa per tutto il tempo necessario al caricamento di un programma, con una eccezione: alcune aziende proteggono i propri dischi commerciali, creando deliberatamente alcuni errori su certi blocchi del disco. Durante o dopo il caricamento questi blocchi vengono controllati dal programma e se risultano rovinati come il programma si aspetta, tutto va liscio, mentre se risultano leggibili – e quindi il disco è un disco "pirata" –, il programma generalmente si autocancella (le protezioni recentissime e più "cattive" arrivano addirittura a riformattare il disco, azzerare la RAM del computer e sbatacchiare talmente la povera testina del drive da metterlo fuori uso fino a che venga riallineato N.d.r.).
Una delle principali cause del disallineamento delle testine è infatti l'uso di dischi commerciali con questo tipo di protezione ad errori: il ripetuto sbattere della testina contro il suo punto di fine corsa provoca alla lunga il disallineamento. Anche il trasporto del drive, con i conseguenti sbatacchiamenti della testina, può provocare il disallineamento o, naturalmente, l'usura dovuta ad un gran numero di accessi ai dischi.
Il test più semplice per valutare lo stato della testina consiste nel mandare in esecuzione il programma "Performance test", contenuto nel disco Commodore fornito insieme al drive stesso, oppure il programmino che viene listato più avanti. Vediamo ora quali sono i passi necessari per effettuare il riallineamento: si controllerà prima l'allineamento rispetto alla traccia 18, caricando una directory molto lunga. Si verificherà in seguito l'allineamento rispetto a tracce più interne e più esterne attraverso il breve programma qui listato.
Siamo pronti per cominciare. Un'ultima raccomandazione è di prestare la massima attenzione a non toccare parti del drive sotto tensione e di leggere più volte che cosa si deve fare prima di operare sul drive.

Disassemblaggio del drive

Ci riferiremo a destra, sinistra, davanti e dietro, pensando che il drive si trovi sul tavolo con la fessura di introduzione dei dischetti esattamente di fronte a voi.
1. Spegnete il drive e staccate tutti i cavi dalla sua parte posteriore.
2. Appoggiate su una superficie piana il drive in posizione capovolta, e rimuovete le quattro viti a croce.
3. Riportate il drive in posizione normale e togliete con delicatezza il suo coperchio superiore.
4. Alcuni tipi di 1541 sono dotati di un ulteriore coperchio metallico, mentre altri ne sono sprovvisti. Nel caso il vostro drive ne sia provvisto, smontatelo, svitandone le due viti di fissaggio sul suo fianco sinistro.
5. Il led verde del drive è collegato alla scheda ora in vista mediante due fili ed una spinetta. Segnatevi l'orientamento della spinetta e staccatela dalla scheda.
6. Rimontate, se presente, il coperchio interno di metallo.
7. Rimuovete la metà inferiore del contenitore plastico del drive, svitandone le sei viti. Estraete quindi il drive.
8. Sistemate il drive in posizione capovolta su di una superficie piana in prossimità del computer. A questo punto dovreste trovare il motore passo-passo e le due viti che lo tengono in posizione (foto 1). Queste viti si trovano all'interno di due aperture e permettono – se allentate – di ruotare di qualche grado il motore stesso.
9. Inserite il cavo di alimentazione nella parte posteriore del drive e la spina in una presa di corrente. **Attenzione!** Da questo momento in poi NON toccate nessuno dei componenti elettrici esposti del drive nella sua parte posteriore. Si può soltanto toccare l'interruttore di alimentazione per accenderlo e spegnerlo. Si può inoltre toccare la gabbia di metallo protettiva che avvolge il drive. Se, per una ragione qualsiasi, questa affermazione risultasse falsa, staccate immediatamente il cavo di alimentazione dalla rete e portate il drive da un buon tecnico: si tratta di un corto circuito interno al drive, ed i suoi problemi sono molto più gravi di un semplice disallineamento della testina.

Procedura di allineamento

1. Collegate il cavo seriale tra drive e computer.
2. Accendete il drive mediante il suo interruttore, ed accendete anche il computer e la TV o il monitor. La luce verde del drive non si accenderà, essendo stata scollegata precedentemente.
3. Pulite lo schermo con SHIFT e CLR/HOME.
4. Inserite un disco, preferibilmente contenente molti programmi e quindi con una directory molto lunga. Ricordate che, essendo il drive capovolto, dovrete inserire capovolto anche il disco.
5. Digitate LOAD"$",8 e premete RETURN. Si accenderà il led rosso e comparirà il messaggio SEARCHING FOR $. Apparirà quindi la scritta LOADING o, se il vostro drive è proprio messo male, otterrete il messaggio FILE NOT FOUND.
Infine apparirà il consueto READY con il cursore. In alcuni casi il drive troverà la directory e cercherà di caricarla senza riuscirci: si otterrà quindi un "blocco" del sistema (il cursore non ritorna sullo schermo). In questo caso spegnete e riaccendete il computer e provate con un altro disco.
6. Durante il caricamento della directory osservate la luce rossa del drive: se rimane accesa per tutto il tempo di caricamento, passate al punto 8. Se la luce rossa si accende e si spegne più volte, o se ottenete il messaggio FILE NOT FOUND, proseguite con il punto 7.
7. Dovete rendere il drive sufficientemente operativo per caricare la directory.
Premete CLR/HOME (senza SHIFT) per portare il cursore nell'angolo superiore sinistro dello schermo, sopra la lettera L di LOAD"$",8. Con il cacciavite a croce allentate (senza svitarle completamente!) le due viti del motore passo-passo (possono essere molto resistenti allo svitamento). Non è necessario spegnere il drive o togliere il disco durante questa operazione. Ruotate il motore passo-passo di circa un millimetro-un millimetro e mezzo in senso orario, e riavvitate a fondo le due viti. Premete RETURN sulla tastiera del computer e ripetete il punto 6.
Se la directory non viene ancora caricata correttamente, ruotate il motore passo-passo in senso antiorario e provate ancora. Può darsi che dobbiate compiere più volte questa operazione, fino a trovare la posizione ottimale del motore passo-passo che consenta un buon caricamento della directory.
8. Se la directory carica bene, ciò significa che l'allineamento sulla traccia 18 è accettabile. La 18 è la traccia più facile da allineare, e sembra dotata di una maggiore tolleranza nei confronti di tarature non ottimali. Le tracce 1 e 35 sono un po' più critiche.
Ritornando al vostro monitor, dopo aver caricato la directory il cursore dovrebbe essere lampeggiante sotto il messaggio READY. Non listate la direc-



tory!
Digitate NEW e premete RETURN.
9. Digitate ora il seguente programmino:

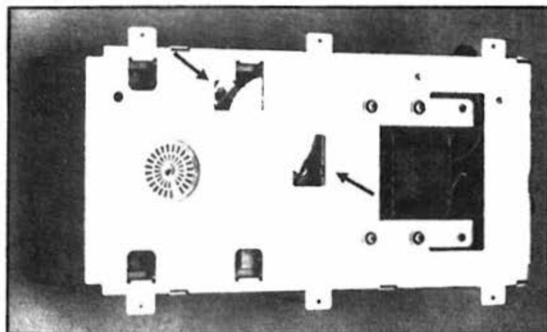
Programma per l'allineamento

```
10 OPEN15,8,15:OPEN5,8,5,"#"
20 PRINT#15,"U1:"5;0;1;1:GOS
   UB60
30 PRINT#15,"U1:"5;0;35;1:GO
   SUB60
40 CLOSE5:CLOSE15
50 END
60 INPUT#15,A,A$,B,C:PRINTA;
   A$;B;C
70 RETURN
```

e dategli il RUN.
10. Il programma controllerà se il blocco 1 delle tracce 1 e 35 risulta leggibile. Appariranno due messaggi sotto l'ultimo comando RUN: se il drive ha un allineamento accettabile, entrambi i messaggi saranno 0 OK 00 00, e potrete passare al punto 13. In caso contrario otterrete un messaggio d'errore, come 21 READ ERROR 01 01 oppure 27 READ ERROR 35 01, o qualcosa di molto simile. Gli ultimi due numeri del messaggio d'errore si riferiscono alla traccia ed al blocco che è risultato illeggibile.
11. Se avete ricevuto un messaggio d'errore al punto 10, dovete effettivamente riallineare la testina. Allentate le due viti del motore passo-passo e ruotatelo di pochissimo in senso orario. Riavvitate a fondo le due viti e ridate il RUN al programma. Se ottenete ancora dei messaggi d'errore, dovete ripetere le operazioni di questo stesso punto, provando a ruotare il motore un po' di più o un po' di meno in senso orario o antiorario. Con il programmino siete in grado di trovare, con molteplici tentativi, la sua posizione giusta.
12. Mentre allineate il drive sulle tracce 1 e 35, non dovete però perdere l'allineamento sulla traccia 18. Per ottenere questo risultato lo schermo del vostro monitor dovebbe aver visualizzato in alto a sinistra l'istruzione LOAD"$",8, subito sotto il SEARCHING FOR $, sotto ancora LOADING e quindi READY. Sotto il READY ci sarà un'istruzione NEW, seguita dalle 7 linee del programma, in fondo alle quali ci sarà un RUN. Ogni volta che spostate la posizione del motore passo-passo dovrete premere CLR/HOME (senza SHIFT) seguito da due RETURN per caricare la directory ed assicurarvi quindi che la traccia 18 sia allineata.
Quindi premere ancora RETURN sull'istruzione NEW, per eliminare la directory dalla memoria del computer, e premere sette volte RETURN per reinserire in memoria le sette linee del programmino. Premendo ancora una volta RETURN sull'istruzione RUN controllerete l'allineamento delle tracce 1 e 35. Con questo sistema dovreste riuscire facilmente, con un po' di tentativi sulla posizione del motore passo-passo, a caricare correttamente la directory e leggere altrettanto correttamente le tracce 1 e 35.
13. Se la directory carica bene, ed ottenete due messaggi 0 OK 00 00 dando il RUN al programma, ciò significa che il drive ha un buon, anche se probabilmente non ottimo, allineamento. Occorre una messa a punto finale su tutte le tracce, per assicurarsi un allineamento perfetto.
14. Digitate quindi il programma listato in fondo a questo articolo, che controlla tutte le tracce del disco. Se ottenete ancora dei messaggi di errore su qualche traccia, ruotate ancora il motore passo-passo di una quantità infinitesima in un senso o nell'altro, fino a che tutto sia OK.

Riassemblaggio del drive

1. Staccate l'alimentazione del drive e il cavo seriale di collegamento al computer.
2. Assicuratevi che le due viti del motore passo-passo siano ben serrate.
3. Inserite il drive nella metà inferiore del suo involucro plastico, ed avvitate le sei viti.
4. Rimuovete, se presente, il coperchio interno di metallo e ricollegate i due cavetti della spia di alimentazione verde. Quindi riavvitate l'eventuale coperchio.
5. Riavvitate la metà superiore dell'involucro plastico, con le sue quattro viti.
6. Ricollegate i cavi nella parte posteriore del drive.
Buon caricamento a tutti!



Programma per la verifica

```
1000 PRINT"{CLR}VERIFICA TRACCE E SETTORI
     "                              :rem 143
1010 PRINT"{ 2 GIU'}INSERISCI UN DISCO NU
     OVO NEL":PRINT"DRIVE E PREMI RETURN"
                                    :rem  16
1020 GETA$:IFA$<>CHR$(13)THEN1020
                                    :rem  88
1030 OPEN15,8,15                    :rem  82
1040 PRINT"{GIU'}FORMATTAZIONE...
     { 2 GIU'}":PRINT#15,"N0:CONTROLLO,92
     ":GOSUB1240:CLOSE15            :rem 192
1050 IFEN>0THENEND                  :rem  92
1060 OPEN15,8,15:OPEN8,8,8,"#"      :rem  84
1070 FORT=1TO35:FORS=0TO20          :rem  98
1080 PRINT"{CLR}CONTROLLO TRACCIA"T;TAB(2
     2)"SETTORE"S                   :rem 184
1090 PRINT#15,"B-P:";8;1            :rem  40
1100 FORI=1TO255:PRINT#8,CHR$(I);:NEXT
                                    :rem  60
1110 PRINT#15,"U2:";8;0;T;S         :rem   5
1120 GOSUB1240                      :rem  11
1130 PRINT#15,"B-P:";8;1            :rem  35
1140 FORI=1TO255:PRINT#8,"-";:NEXT
                                    :rem  22
1150 PRINT#15,"U1:";8;0;T;S         :rem   8
1160 GOSUB1280                      :rem  19
1170 PRINT#15,"B-P:";8;1            :rem  39
1180 FL=0:GL=0:FORI=1TO255          :rem  26
1190 GET#8,A$:IFA$<>CHR$(I)THENFL=1
                                    :rem   9
1200 NEXT:IFFL=1THENGOSUB1270       :rem  68
1210 IF(T>17ANDS>17)OR(T>24ANDS>16)OR(T>3
     0ANDS>15)THENS=20              :rem 241
1220 NEXT:NEXT:PRINT"{ 3 GIU'}{RVS}ATTENZ
     IONE: RIFORMATTARE":PRINT"{RVS}PRIMA
      DELL'USO!"                    :rem 142
1230 CLOSE8:CLOSE15:END             :rem 155
1240 INPUT#15,EN,EM$,ET,ES          :rem  10
1250 IFEN=0THENRETURN               :rem 102
1260 GL=1:PRINT"{GIU'}ERRORE SCRITTURA:"E
     N;EM$;ET;ES:RETURN             :rem 245
1270 PRINT"BLOCCO DIFETTOSO:TRACCIA"T;TAB
     (22)"SETORE"S"{GIU'}":RETURN
                                    :rem  34
1280 INPUT#15,EN,EM$,ET,ES          :rem  14
1290 IFEN=0THENRETURN               :rem 106
1300 IFGL=0THENPRINT                :rem  15
1310 PRINT"ERRORE LETTURA:"EN;EM$;ET;ES:R
     ETURN                          :rem   5
```



Ultrawedge

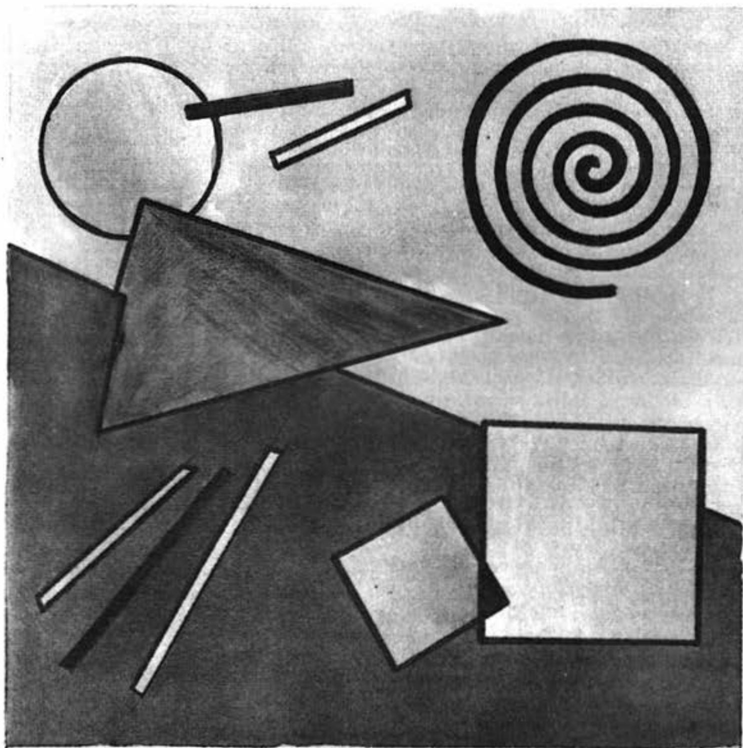
Prima parte

di **J. Klitzing**
trad. ed adatt.
di **S. Colombo**

Una delle più perfezionate utility per C64 è ora disponibile per tutti i lettori di SUPERCOMMODORE: 37 nuovi comandi gestibili da BASIC più un monitor con altri 31 comandi, il tutto in un solo programma, che lascia inalterata la quantità di memoria disponibile per la programmazione in BASIC!

ULTRAWEDGE è un programma interamente in linguaggio macchina, lungo quasi 10 Kbyte, che si posiziona in zone della memoria RAM inaccessibili al BASIC, permettendo così di sfruttare interamente tutta la memoria disponibile dalla locazione 2049 fino alla 40960.
I comandi messi a disposizione da ULTRAWEDGE sono orientati principalmente all'uso del disk drive, e forniscono un tool completo e professionale per chi usa intensamente questa periferica. Tutti i comandi sono utilizzabili esclusivamente in modo diretto, e non attraverso linee di programma; se non altrimenti specificato, i numeri contenuti nei comandi possono essere sia in notazione decimale che esadecimale e se si usa la notazione esadecimale, il numero deve essere preceduto dal simbolo $ (ad esempio, 42 decimale oppure $2A esadecimale).
I comandi che scrivono dati in appositi buffer nella memoria del C64 non disturbano i programmi BASIC, né sovrascrivono variabili o stringhe: i buffer vengono infatti allocati da ULTRAWEDGE in aree di memoria RAM, che non vengono utilizzate dai programmi BASIC.
Vediamo ora quali sono e che funzioni hanno i comandi di ULTRAWEDGE:

• questo comando legge il canale degli errori del drive e ne stampa il messaggio sul video. È utile per sapere che cosa non ha funzionato quando la spia rossa del drive lampeggia, segnalando un errore.



•A generazione automatica delle linee e rinumerazione dei programmi; può avere tre forme:

•A,n

inizia la generazione automatica di numeri di linea a partire dal valore n (n può variare da 1 a 63999). Dopo aver digitato una linea di programma, e premuto RETURN, verrà generato il numero di linea successivo ed il cursore alla sua destra attenderà la digitazione della linea seguente. Questa funzione verrà automaticamente disabilitata quando venga eseguita una qualsiasi funzione che non sia di editing, compreso il LIST, RUN, PEEK, POKE, ecc., oppure può essere disabilitata digitando •A senza farla seguire dalla virgola e dal numero. La seconda forma di questo comando è:

•A,n,m

e serve per rinumerare i programmi BASIC; n è il numero di linea più basso dal quale desiderate che inizi la numerazione del programma, ed m è l'incremento tra i numeri di linea. Ovviamente, questa funzione rinumera anche i GOTO, GOSUB, IF/THEN, ecc. Se possedete qualche altro tool in linguaggio macchina che esegue questa funzione, rimarrete senz'altro stupiti dalla velocità di esecuzione di ULTRAWEDGE:

per rinumerare completamente un lungo programma in BASIC sono necessari soltanto pochissimi secondi. Dopo aver dato questo comando comparirà il messaggio "In corso..." e verrà visualizzata una serie di puntini per dimostrare che ULTRAWEDGE sta lavorando nella rinumerazione del vostro programma.
La terza forma del comando è quella citata prima:

*A

e serve per disabilitare la funzione di generazione automatica dei numeri di linea. Premere SHIFT+RETURN e quindi digitare questo comando seguito da RETURN.

***BR,$n,$m** legge dal disco il contenuto di un settore, specificato dal numero n della traccia e dal numero m del settore (in notazione esadecimale). Il contenuto del settore o blocco specificato viene caricato in un buffer all'interno del C64, viene visualizzata la locazione di inizio del buffer e viene abilitato il SUPERMONITOR 2.0, in modo da poter visualizzare i dati (vedi più avanti i comandi disponibili con il SUPERMONITOR 2.0).

***BW,$n,$m** scrive sul disco il contenuto del buffer, in un settore specificato dal numero n della traccia e dal numero m del settore (in notazione esadecimale). Questo comando viene comunemente usato dopo *BR, per riscrivere sul disco i dati contenuti in un certo modo, dopo averli modificati secondo le proprie esigenze: con *BR si carica nel buffer il contenuto di un certo blocco, con il SUPERMONITOR 2.0 lo si modifica e con *BW lo si riscrive sul disco. Questo comando è molto potente e se usato senza sapere esattamente che cosa si sta facendo, può rovinare irreparabilmente i dati contenuti in un disco. Si consiglia quindi, se si intende operare sul disco con questo comando, di creare sempre una copia di back-up del disco stesso. Per questa ragione ULTRAWEDGE chiede conferma prima di eseguire una "block-write": se si risponde alla richiesta di conferma premendo il tasto S, la scrittura verrà effettuata, mentre la pressione di un qualsiasi altro tasto provocherà l'uscita dall'opzione.

***C:nomefile** questo comando serve per copiare singoli file su disco: il file viene letto dal disco in un buffer all'interno del C64, quindi verrà richiesto l'inserimento del disco destinazione per la riscrittura del file stesso che avverrà alla pressione di RETURN. Il disco destinazione deve essere stato precedentemente formattato: per file molto lunghi saranno necessari più passaggi per copiare l'intero file. Questa opzione permette di copiare sia programmi in BASIC che in linguaggio macchina (ad esempio, *C:ULTRAWEDGE copierà su un altro disco il programma di nome "ULTRAWEDGE"). Vengono copiati anche file sequenziali ed utente, ma non i relativi. Questi ultimi possono comunque essere copiati con l'opzione di back-up dell'intero disco. Questo comando può servire anche per recuperare file che siano stati precedentemente cancellati dal disco con il comando SCRATCH del DOS, o con *S. È sufficiente specificarne il nome e in questo caso il file non verrà copiato, ma solo recuperato all'interno del disco.

***C:nomefile,tipo** è usato per cambiare il tipo dei file, o per recuperare file cancellati in un particolare tipo di file. Per esempio, *C:ULTRAWEDGE,S cambierà il file di nome "ULTRAWEDGE" in un file sequenziale; *C:DATI,P cambierà il file sequenziale di nome "DATI" in file programma.

C copia l'intero disco. Il numero dei passaggi richiesti dipende dalla quantità di dati presente sul disco stesso, in quanto vengono copiati soltanto i blocchi allocati nella BAM. Il disco destinazione deve essere stato precedentemente formattato. Usando invece il comando *C** si avrà la copia di tutti i blocchi del disco, siano essi allocati nella BAM oppure no.

***D** visualizza sullo schermo la directory del disco, senza disturbare il programma eventualmente in memoria. La pressione del tasto CTRL rallenterà, come al solito, il listing, mentre la pressione di un qualsiasi altro tasto provocherà il suo "congelamento".

***F** visualizza l'ammontare di memoria libera per i programmi BASIC, in decimale ed in esadecimale.

***I** inizializza il drive. Equivale all'istruzione BASIC OPEN15,8,15: PRINT # 15,"0:I":CLOSE15

***L:nomefile** visualizza l'indirizzo di partenza di un file programma su disco, sia in decimale che in esadecimale. Questo comando è utile per conoscere le SYS di attivazione di molti programmi in linguaggio macchina, o per sapere in quale area caricare i programmi attraverso il SUPERMONITOR 2.0.

***M** entra in SUPERMONITOR 2.0, il programma assemblatore e disassemblatore di linguaggio macchina. SUPERMONITOR 2.0, già compreso in ULTRAWEDGE, differisce da SUPERMONITOR (pubblicato sul numero di Gennaio della rivista) per due caratteristiche fondamentali: mentre la versione originaria occupava la memoria da $C000 a $CFFF, la versione 2.0 risiede "sotto" l'interprete BASIC, da $B000 a $BFFF, permettendo l'analisi di un maggior numero di programmi in linguaggio macchina, e il comando "P" di SUPERMONITOR 2.0 attiva o meno la stampante (periferica 4) sul bus seriale invece che sulla porta RS-232. Inoltre, SUPERMONITOR 2.0 è interfacciato con molte delle subroutine di ULTRAWEDGE. I comandi di SUPERMONITOR 2.0 vengono descritti separatamente dopo quelli di ULTRAWEDGE.

***N:nomedisco,id** formatta un disco, ed equivale all'istruzione BASIC OPEN15,8:PRINT # 15,"NO:nomedisco,id":CLOSE15

***P:nomefile** stampa un file sequenziale sullo schermo o sulla stampante (periferica 4 sul bus seriale). La pressione di CTRL, mentre la visualizzazione avviene sul video, provoca un rallentamento; la pressione di qualsiasi altro tasto provoca una pausa. Mentre ci si trova in pausa, premendo "X" si esce dall'opzione; la pressione di qualsiasi altro tasto ne riprenderà l'esecuzione.

***Q** disabilita ULTRAWEDGE. Per riabilitarlo dopo aver usato questo comando digitare SYS49152.

***R:nuovonome=vecchionome** cambia il nome ad un file su disco, ed equivale all'istruzione BASIC OPEN 15,8,15:PRINT # 15,"R0:nuovonome=vecchionome":CLOSE15.

***S:nomefile** cancella un file su disco, ed equivale all'istruzione BASIC OPEN15,8,15:PRINT # 15,"20:nomefile":CLOSE15.

***U** recupera dalla memoria BASIC un programma precedentemente cancellato con l'istruzione NEW. Esegue inoltre un CLR.

***V** esegue la validate sul disco, ed equivale all'istruzione BASIC

OPEN15,8,15:PRINT # 15,"V0": CLOSE15.

***X** provoca la stampa di ciò che è visualizzato sullo schermo. La linea contenente il comando viene cancellata, in modo da non avere anche la stampa dei caratteri "*X". Si consiglia di non dare questo comando sulla linea più bassa dello schermo, in quanto così operando si avrebbe uno scroll di tutto lo schermo in senso verticale. Questo comando non permette la stampa di schermi in alta risoluzione e di caratteri in campo inverso (che vengono comunque stampati normalmente). Permette invece la stampa di caratteri sia maiuscoli che minuscoli e di tutti i caratteri grafici del set standard.

***!** esegue la routine di "cold start" del C64, ed equivale a una SYS64738.

***#n** regola ULTRAWEDGE in modo che possa comunicare con un disk drive il cui numero di periferica sia diverso da 8. Il numero n di periferica può andare da 8 a 31 e deve essere specificato in decimale.

***=n** modifica via software il numero di periferica del drive, portandolo sul numero n, e regola contemporaneamente ULTRAWEDGE per la comunicazione con il disk drive numero n. Anche qui n può variare da 8 a 31 e deve essere specificato in decimale.

***+n** modifica il colore del bordo, portandolo al colore il cui codice è n. Per un elenco dei codici di colore disponibili sul C64 fare riferimento al manuale in dotazione al computer.

***–n** modifica il colore del fondo, portandolo al colore il cui codice è n. Per un elenco dei codici di colore disponibili sul C64 fare riferimento al manuale in dotazione al computer.

***£n** modifica il colore del testo, portandolo al colore il cui codice è n. Per un elenco dei codici di colore disponibili sul C64, fare riferimento al manuale in dotazione al computer.

***?** visualizza un elenco abbreviato dei comandi di ULTRAWEDGE.

***/nomefile,n** carica un programma da disco a partire dall'indirizzo decimale o esadecimale n. Se n non è specificato, il caricamento avviene a partire dall'inizio del BASIC. Equivale all'istruzione LOAD"nomefile",8.

***,nomefile,n** carica un programma da disco a partire dall'indirizzo decimale o esadecimale n. Se n non è specificato, il caricamento avviene nella stessa area di memoria dalla quale il file era precedentemente stato salvato. Equivale all'istruzione LOAD"nomefile",8,1.

***↑nomefile,n** carica da disco un programma a partire dalla locazione n decimale o esadecimale. Se n non è specificato, il caricamento avviene a partire dall'inizio del BASIC, e viene forzato un RUN, che manda automaticamente in esecuzione il programma.

***←nomefile,n,m** salva su disco un programma il cui indirizzo iniziale è n, ed il cui indirizzo finale +1 è m. I parametri n ed m possono essere sia decimali che esadecimali. Se non specificati, il salvataggio avviene a partire dall'inizio del BASIC, ed il comando equivale all'istruzione SAVE"nomefile",8.

***:nomefile,n** carica da disco un file sequenziale, a partire dall'indirizzo decimale o esadecimale n (se specificato). I dati del sequenziale possono essere letti attraverso SUPERMONITOR 2.0.

***;nomefile,n,m** salva su disco un file sequenziale precedentemente caricato, definito dagli indirizzi - opzionali - iniziale n e finale m–1; n ed m possono essere sia decimali che esadecimali.

@ nomefile,n carica da disco un file utente (USR) a partire dall'indirizzo decimale o esadecimale n (se specificato). I dati del file utente possono essere letti attraverso SUPERMONITOR 2.0.

****nomefile,n,m** salva su disco un file utente (USR) precedentemente caricato, definito dagli indirizzi - opzionali - iniziale n e finale m–1; n ed m possono essere sia decimali che esadecimali.

I comandi disponibili con il SUPERMONITOR 2.0 sono dettagliatamente descritti sul numero di Gennaio '85 di questa rivista. Ne forniamo comunque un breve riassunto:

ASSEMBLER

.A (es: .A 401F LDA $0314)

L' input iniziale di questo comando richiede un indirizzo di partenza in esadecimale. Una volta digitata una linea l'assembler visualizzerà la lettera di comando A seguita dall'indirizzo dell' istruzione seguente. Le istruzioni ASSEMBLY sono codici mnemonici di tre caratteri seguiti da un campo opzionale per l'operando. I dati per l' operando devono essere in esadecimale e prefissati dal simbolo $. I dati con indirizzamento immediato devono inoltre essere prefissati dal simbolo #. Tutti i riferimenti agli indirizzi sono specificati in esadecimale e sono da intendersi assoluti. I salti relativi sono calcolati dall' assembler, usando la differenza tra l' obiettivo e l' indirizzo corrente del salto. Per uscire dall' assembler premere RETURN dopo la visualizzazione della lettera A.

BREAK SET

.B (es: .B 2000 0010)

Questo comando permette di fermare l'esecuzione di un programma in linguaggio macchina dopo che un indirizzo specificato sia occorso per un prefissato numero di volte. L'esecuzione del programma in linguaggio macchina deve essere iniziata con il comando QUICK TRACE. Nell'esempio l'esecuzione del programma in linguaggio macchina viene interrotta alla sedicesima volta che viene eseguita l'istruzione contenuta nella locazione $2000.

COMPARE MEMORY

.C (es: .C 2000 2FFF C000)

Viene eseguito il confronto tra due blocchi di memoria, avendo come output gli indirizzi delle locazioni del primo blocco che contengono dati non coincidenti con le corrispondenti locazioni del secondo blocco. Nell'esempio il primo blocco di memoria da $2000 a $2FFF viene confrontato con il secondo blocco da $C000 a $CFFF.

DISASSEMBLER

.D (es. .D 5015 501C)

Un blocco di programma in linguaggio macchina può essere disassemblato e visualizzato, specificando la locazione iniziale e finale del blocco stesso. Il tasto RUN/STOP può essere usato per interrompere il listato. Se viene specificato un solo indirizzo, verrà disassemblata solo l' istruzione contenuta in quella locazione. Il linguaggio macchina può essere "editato", usando i tasti cursore per muoversi e modificare i byte. La pressione del tasto RETURN immetterà in memoria byte così modificati.

EXIT BASIC

.E

Questo comando resetta i vettori delle interruzioni IRQ e BRK, riporta il buffer cassette a $033C e ritorna al BASIC. Il comando E deve essere usato per uscire dal SUPERMONITOR 2.0 quando si intendano usare in ambiente BASIC i comandi LOAD, SAVE o VERIFY. Si usi sempre *M per rientrare nel SUPERMONITOR 2.0.

FILL MEMORY

.F (es: .F 2000 3FFF 00)

Questo comando riempie un blocco di memoria con il byte specificato. La memoria viene scritta dal basso verso l'alto, e nessun controllo viene fatto sulla scrittura.

GO

.G (es: .G 2000)

Viene mandato in esecuzione un programma in linguaggio macchina che inizia alla locazione specificata. Nell'esempio il programma inizia dall'indirizzo esadecimale $2000.

HUNT

.H (es: .H 1000 5FFF 01 02 03)

Un blocco di memoria specificato da un indirizzo inferiore e superiore viene letto dal basso verso l'alto alla ricerca di un massimo di 32 caratteri o byte specificati. Viene visualizzato l'indirizzo di ogni locazione in cui sia contenuto uno dei caratteri o byte specificati. La ricerca può essere interrotta, premendo RUN/STOP durante la visualizzazione degli indirizzi.

JUMP

.J (es: .J 3000)

Viene chiamata la subroutine iniziante alla locazione specificata, rimanendo in ambiente SUPERMONITOR 2.0. La subroutine deve terminare con un'istruzione RTS (ReTurn from Subroutine), che provoca il ritorno alla sezione input comandi del SUPERMONITOR 2.0.

LOAD

.L (es: .L 4000 "PROVA" 08)

Cerca e, se trovato, carica in memoria a partire dall' indirizzo $4000 il file chiamato PROVA dalla periferica numero 8. Se non viene specificato il numero di periferica, viene assunto per default il valore 1, corrispondente all' unità a cassette.

MEMORY DISPLAY

.M (es: .M CE78 CE88)

Questo comando visualizza il contenuto della memoria otto byte alla volta, seguiti dalla loro codifica ASCII. I byte seguenti l' indirizzo possono essere modificati muovendo il cursore sopra di essi e digitando i nuovi byte. I byte modificati vengono inseriti in memoria alla pressione del tasto RETURN.

NEW LOCATER

.N (es: .N 2000 2003 6000 C000 CFFF)

L'esempio modifica tutte le istruzioni di tre byte nell'area di memoria da $2000 a $2003, aggiungendo il valore $6000 all'indirizzo assoluto nei due byte seguenti l'opcode dell'istruzione. Ogni indirizzo assoluto trovato all'esterno dell'intervallo da $C000 a $CFFF non viene rilocato. Inoltre, se viene incontrato un opcode non valido, l'operazione termina con la visualizzazione del disassemblato dell'opcode non valido.

OFFSET O BRANCH CALCULATE

.O (es: .O 1004 1000 FA)

Calcola l'entità del salto in istruzioni branch. Il primo indirizzo è per la locazione contenente l'opcode del branch, mentre il secondo indirizzo contiene il numero di locazione di arrivo del salto.

PRINT SWITCHER

.P

Se l'output selezionato (default) è lo schermo, questo comando lo passa sul bus seriale (periferica #4). Se l'output selezionato non è lo schermo, lo ripassa a quest' ultimo, lasciando aperto il canale seriale fino a che il suo buffer si sia svuotato.

QUICK TRACE

.Q (es: .Q 4000)

Ogni istruzione viene eseguita come con il comando Walk, ma senza alcun output. L'esecuzione del programma in linguaggio macchina non avviene alla massima velocità. La pressione del tasto RUN/STOP ne interrompe l'esecuzione, visualizzando il carattere S* seguito dall'immagine memorizzata dei registri.

REGISTER DISPLAY

.R

Visualizza il contenuto dei registri, come quando si entra per la prima volta in SUPERMONITOR 2.0. L'immagine dei registri può essere modificata, posizionandosi con il cursore sui valori di registro che si desidera modificare e digitando i nuovi valori. Le modifiche vengono memorizzate con la pressione del tasto RETURN.

SAVE

.S (es: .S 4000 5000 "PROVA" 08)

Salva - nell'esempio - il contenuto della memoria fino all' indirizzo $5000 (escluso) sulla periferica numero 8, che è il drive 1541. Se il numero di periferica non viene specificato, viene assunto per default il valore 1 corrispondente all' unità a cassette.

TRANSFER MEMORY

.T (es: .T 4000 4FFF 6000)

Trasferisce una copia dei dati contenuti nell'area di memoria da $4000 a $4FFF nelle locazioni da $6000 a $6FFF.

VERIFY

.V (es: .V 4000 "PROVA" 08)

Cerca e, se esiste, verifica con la memoria il file "PROVA" sulla periferica numero 8 (disk drive). Se non specificato, il numero di periferica viene assunto per default di valore 1 (unità a cassette).

WALK CODE

.W (es: .W 4000)

Una singola istruzione viene eseguita, viene generato un segnale IRQ e la nuova immagine macchina viene memorizzata e visualizzata come SR, AC, XR, YR, SP seguiti da indirizzo, codice macchina e disassemblato della successiva istruzione da eseguire. La pressione del tasto RUN/STOP interrompe l'esecuzione del Walk. La pressione del tasto J provoca il proseguimento nell'esecuzione del programma in linguaggio macchina alla massima velocità. La pressione di qualunque altro tasto durante il Walk provoca l'esecuzione dell'istruzione successiva.

EXIT

.X

Ritorna in ambiente BASIC. Questo comando permette di operare in ambiente BASIC, ma tratta l'esecuzione delle interruzioni BRK ancora attraverso il sistema di interruzione software del SUPERMONITOR 2.0.

CONVERSIONE ASCII

." (es: ."B)

Un carattere ASCII, grafico o di controllo, viene digitato al fine di ottenere il suo valore esadecimale, decimale e binario.

CONVERSIONE DECIMALE

.# (es: .#16706)

Un numero decimale viene digitato al fine di ottenere il suo valore esadecimale, i caratteri ASCII dei due byte ed il valore binario.

CONVERSIONE BINARIA

.% (es: .%011010)

Un numero binario viene digitato al fine di ottenere il suo valore esadecimale, decimale ed i caratteri ASCII dei due byte costituenti il numero stesso.

CONVERSIONE ESADECIMALE

.$ (es: .$4142)

Un numero esadecimale viene digitato al fine di ottenere il suo valore decimale, i caratteri ASCII dei due byte ed il valore binario. I tasti di cursore su/cursore giù possono essere usati per provocare uno scorrimento dello schermo con visualizzazione dei numeri esadecimali superiori o inferiori.

CHECKSUM

.& (es: .& C000 CFFF A500)

I dati nel blocco di memoria da $C000 a $CFFF compreso sono sommati byte per byte e visualizzati.

SEGNALE ACUSTICO

.(

Abilita il segnale acustico di fine comando. Un suono continuo viene generato alla fine dell'esecuzione del comando successivo. Il suono può essere interrotto premendo RETURN. Non viene generato alcun suono, se esiste una condizione di errore mentre si digita il successivo comando. Il comando: .) disabilita il segnale acustico.

ADDIZIONE

.+ (es: .+ 1111 2222 3333)

Due numeri esadecimali sono digitati per ottenere la loro somma in modulo 16.

SOTTRAZIONE

.– (es: .- 3333 1111 2222)

Due numeri esadecimali sono digitati ed il secondo viene sottratto dal primo per ottenerne la differenza. La sottrazione è calcolata con aritmetica in complemento a due.

DIRECTORY

.>

Digitando > seguito da RETURN si ottiene la visualizzazione della directory del disco presente nel drive.

Note per il caricamento da cassetta

A causa della sua lunghezza ULTRAWEDGE viene pubblicato in due parti: la prima su questo numero e la seconda sul numero del mese prossimo. Sulla cassetta allegata a questo numero della rivista è tuttavia presente l'intero programma. Per caricare il programma è sufficiente digitare:

LOAD"ULTRAWEDGE",1,1
(RETURN)

Alla fine del caricamento del programma verrà visualizzato un messaggio di ?LOAD ERROR. Ciò è perfettamente normale, ed il programma è stato invece caricato con successo.

Il messaggio di errore deriva infatti da una particolarità di ULTRAWEDGE: quella di risiedere in parte sotto la ROM dell'interprete BASIC. Il sistema di caricamento Commodore è composto da una prima fase, in cui avviene il caricamento vero e proprio, ed una seconda fase, in cui viene comparato con la memoria ciò che il computer legge sul nastro. Poiché la verifica avviene con la ROM del BASIC e non con la RAM sottostante, si ottiene questo messaggio d'errore. In realtà il programma è stato caricato con successo. All'apparizione del messaggio di LOAD ERROR digitate NEW e premete RETURN. Digitate quindi:

SYS49152 (RETURN)

ULTRAWEDGE è ora in funzione, anche se nulla sembra succedere. Verificatelo digitando il comando *?, per visualizzare l'elenco dei comandi. Avete a vostra disposizione quasi settanta nuovi comandi da utilizzare secondo i vostri desideri.

Note per la digitazione del programma

Chi non possiede la cassetta allegata alla rivista dovrà utilizzare, per la digitazione del codice macchina, la versione speciale qui riportata di MLX. L'usuale MLX non è infatti in grado di salvare la RAM sotto la ROM del C64, né di leggerla quando si usi il comando SHIFT-D.

Il listato del codice macchina di ULTRAWEDGE è diviso in due parti: quella qui pubblicata e quella che verrà pubblicata sul prossimo numero. La procedura da seguire, anche se può apparire un po' complicata, è la seguente:

1. Digitare la versione speciale di MLX e salvarla su disco o nastro.
2. Spegnere e riaccendere il C64, caricare la versione speciale di MLX e rispondere alle sue domande con:

indirizzo iniziale: 40960
indirizzo finale : 42555

3. Iniziare quindi a digitare la prima parte del codice macchina di ULTRAWEDGE e salvarla su nastro o disco con il nome "PARTE1".

Riprendendo tra un mese la digitazione della seconda parte, occorrerà seguire la seguente procedura:
1. Spegnere e riaccendere il C64, caricare la versione speciale di MLX e rispondere alle sue domande con:

indirizzo iniziale: 45056
indirizzo finale : 53251

2. Iniziare quindi a digitare la seconda parte del codice macchina di ULTRAWEDGE e salvarla su nastro o disco con il nome "PARTE2".

Si noterà che le due parti non sono contigue (la prima termina in 42555, mentre la seconda inizia in 45056). Per renderle contigue, cioè poterle caricare entrambe come un unico programma, occorrerà compiere le seguenti operazioni:
1. Sincerarsi di avere registrate su nastro o disco le due parti del programma, con i nomi di "PARTE1" e "PARTE2".
2. Spegnere e riaccendere il C64, caricare la versione speciale di MLX e rispondere alle sue domande con:

indirizzo iniziale: 40960
indirizzo finale : 53251

3. Alla comparsa del numero 40960, seguito dal cursore di MLX, premere contemporaneamente SHIFT e L: caricare quindi "PARTE1" da nastro o disco.
4. Alla ricomparsa del numero 40960, seguito dal cursore di MLX, premere nuovamente SHIFT e L e caricare "PARTE2" da nastro o disco.
5. Ricomparirà ancora, dopo il caricamento, il numero 40960 seguito dal cursore di MLX. A questo punto premere contemporaneamente i tasti SHIFT ed S e salvare il programma su nastro o disco con il nome di "ULTRAWEDGE".

Ora vi troverete su nastro o disco l'intero programma, e potrete usare per il suo caricamento la procedura precedentemente descritta per i possessori di cassetta.
Si noti che utilizzando il disk drive non si ottiene alcun messaggio di errore dopo il caricamento del programma.

MLX: versione speciale per Ultrawedge

```
90 REM LINEE AGGIUNTE O MODIFICATE RISPET
   TO ALLA VERSIONE 2.0:              :rem 251
95 REM 160,170,180,190,210,240,420,880,89
   0                                  :rem 20
96 REM 900,920,930,1040,1060,1070,1160,11
   70,1180,1190                       :rem 41
100 PRINT"{CLR}{CYN}";CHR$(142);CHR$(8);:
    POKE53280,0:POKE53281,0           :rem 71
110 POKE788,52:REM DISABILITA RUN/STOP
                                      :rem 89
120 PRINT"{RVS}{ 40 SPAZI}";          :rem 177
130 PRINT"{RVS}{ 15 SPAZI}{DES}{OFF}[<*>]
    £{RVS}{DES} {DES}{ 2 SPAZI}[<*>]{OFF}
    [<*>]£{RVS}£{RVS}{ 13 SPAZI}";
                                      :rem 251
140 PRINT"{RVS}{ 15 SPAZI}{DES}[<N>][<H>]
    {DES} { 2 DES} {OFF}£{RVS}£[<*>]{OFF}
    [<*>]{RVS}{ 13 SPAZI}";           :rem 221
150 PRINT"{RVS}{ 40 SPAZI}"           :rem 121
160 PRINT"{ 2 GIU'}{PUR}VERSIONE SPECIALE
     2.5 PER ULTRAWEDGE{ 2 SPAZI}
    { 3 GIU'}"                        :rem 122
170 DATA162,54,134,1,174,251,0,32,216,255
    ,166,55,134,1,96                  :rem 34
180 DATA162,54,134,1,174,0,0,134,251,162,
    55,134,1,96                       :rem 33
190 LM=6000:FORI=LMTOLM+28:READD:POKEI,D:
    NEXT                              :rem 124
200 PRINT"[<5>]{ 2 SU} INDIRIZZO DI PARTE
    NZA{ 2 SPAZI}";:INPUTS:F=1-F:C$=CHR$(
    31+119*F)                         :rem 84
210 IFS<256ORS>53247THENGOSUB1160:GOTO200
                                      :rem 1
220 PRINT:PRINT:PRINT                 :rem 175
230 PRINT"[<5>]{ 2 SU} INDIRIZZO CONCLUSI
    VO{ 3 SPAZI}";:INPUTE:F=1-F:C$=CHR$(3
    1+119*F)                          :rem 92
240 IFE<256ORE>53251THENGOSUB1160:GOTO230
                                      :rem 230
250 IFE<STHENPRINTC$;"{RVS}INDIRIZZO CONC
    LUSIVO<INDIRIZZO INIZIALE"        :rem 200
260 IFE<STHENGOSUB1080:GOTO230        :rem 123
270 PRINT:PRINT:PRINT                 :rem 180
280 PRINT"{CLR}";CHR$(14):AD=S:POKEV+21,0
                                      :rem 232
290 A=1:PRINTRIGHT$("0000"+MID$(STR$(AD),
    2),5);":";:FORJ=ATO6              :rem 234
300 FORJ=ATO6                         :rem 27
310 GOSUB570:IFN=-1THENJ=J+N:GOTO310
                                      :rem 226
320 IFN=-211THEN780                   :rem 62
330 IFN=-204THEN940                   :rem 63
340 IFN=-206THENPRINT:INPUT"{GIU'} INSERI
    RE IL NUOVO INDIRIZZO";ZZ         :rem 15
350 IFN=-206THENIFZZ<SORZZ>ETHENPRINT"
    {RVS}ESCE DAL CAMPO DI VALORI INDICAT
    O"                                :rem 104
360 IFN=-206THENIFZZ<SORZZ>ETHENGOSUB1080
    :GOTO340                          :rem 211
370 IFN=-206THENAD=ZZ:PRINT:GOTO290
                                      :rem 243
380 IFN<>-196THEN450                  :rem 135
390 PRINT:INPUT"LISTATO:DA";F:PRINT"
    { 9 SPAZI}A";:INPUTT              :rem 188
400 IFF<SORF>EORT<SORT>ETHENPRINT"MINIMO"
    ;S;" MASSIMO";E;"![<5>]":GOTO390
                                      :rem 148
410 FORI=FTOTSTEP6:PRINT:PRINTRIGHT$("000
    0"+MID$(STR$(I),2),5);":";        :rem 26
420 FORK=0TO5:GOSUB1170:N=PEEK(251):PRINT
    RIGHT$("00"+MID$(STR$(N),2),3);",";
                                      :rem 154
430 GETA$:IFA$>""THENPRINT:PRINT:GOTO290
                                      :rem 29
440 NEXTK:PRINTCHR$(20);:NEXTI:PRINT:PRIN
    T:GOTO290                         :rem 54
450 IFN<0THENPRINT:GOTO290            :rem 172
460 A(J)=N:NEXTJ                      :rem 196
470 CKSUM=AD-INT(AD/256)*256:FORI=1TO6:CK
    SUM=(CKSUM+A(I))AND255:NEXT       :rem 206
480 PRINTCHR$(18);:GOSUB570:PRINTCHR$(146
    );                                :rem 100
490 IFN=-1THENA=6:GOTO300             :rem 254
500 PRINTCHR$(20):IFN=CKSUMTHEN530
                                      :rem 116
510 PRINT:PRINT"{RED}LA LINEA E' STATA IN
    SERITA IN MANIERA"                :rem 156
520 PRINT"ERRATA. RIPETERE[<5>]":PRINT:GO
    SUB1080:GOTO290                   :rem 37
530 GOSUB1120                         :rem 220
540 FORI=1TO6:POKEAD+I-1,A(I):NEXT:POKE54
    272,0:POKE54273,0                 :rem 227
550 AD=AD+6:IFAD<ETHEN290             :rem 219
560 GOTO780                           :rem 115
570 N=0:Z=0                           :rem 88
580 PRINT"[<£>]";                     :rem 81
590 GETA$:IFA$=""THEN590              :rem 95
600 AV=-(A$="M")-2*(A$=",")-3*(A$=".")-4*
    (A$="J")-5*(A$="K")-6*(A$="L")
                                      :rem 32
610 AV=AV-7*(A$="U")-8*(A$="I")-9*(A$="O"
    ):IFA$="H"THENA$="0"              :rem 125
620 IFAV>0THENA$=CHR$(48+AV)          :rem 125
630 PRINTCHR$(20);:A=ASC(A$):IFA=13ORA=44
    ORA=32THEN720                     :rem 216
640 IFA>128THENN=-A:RETURN            :rem 133
650 IFA<>20THEN680                    :rem 20
660 GOSUB740:IFI=1ANDT=44THENN=-1:PRINT"
    {SIN} {SIN}";:GOTO740             :rem 169
670 GOTO570                           :rem 114
680 IFA<48ORA>57THEN580               :rem 110
690 PRINTA$;:N=N*10+A-48              :rem 111
700 IFN>255THENA=20:GOSUB1080:GOTO650

                                      :rem 238
710 Z=Z+1:IFZ<3THEN580                :rem 67
720 IFZ=0THENGOSUB1080:GOTO570        :rem 118
730 PRINT",";:RETURN                  :rem 236
740 S%=PEEK(209)+256*PEEK(210)+PEEK(211)
                                      :rem 145
750 FORI=1TO3:T=PEEK(S%-I)            :rem 63
760 IFT<>44ANDT<>58THENPOKES%-I,32:NEXT
                                      :rem 198
770 PRINTLEFT$("{ 3 SIN}",I-1);:RETURN
                                      :rem 14
780 PRINT"{CLR}{RVS}*** SAVE ***
    { 3 GIU'}"                        :rem 243
790 PRINT"{ 2 GIU'}PREMERE {RVS}RETURN
    {OFF} PER USCIRE DAL SAVE{GIU'}"
                                      :rem 106
800 F$="":INPUT"{GIU'}NOME DEL FILE";F$:I
    FF$=""THENPRINT:PRINT:GOTO290
                                      :rem 48
810 PRINT:PRINT"{ 2 GIU'}{RVS}N{OFF}ASTRO
     O {RVS}D{OFF}ISCO: (N/D)"        :rem 127
820 GETA$:IFA$<>"N"ANDA$<>"D"THEN820
                                      :rem 28
830 DV=1-7*(A$="D"):IFDV=8THENF$="0:"+F$:
    OPEN15,8,15,"S"+F$:CLOSE15        :rem 211
840 T$=F$:ZK=PEEK(53)+256*PEEK(54)-LEN(T$
    ):POKE782,ZK/256                  :rem 2
```

```
850 POKE781,ZK-PEEK(782)*256:POKE780,LEN(
    T$):SYS65469                  :rem  107
860 POKE780,1:POKE781,DV:POKE782,1:SYS654
    66                            :rem   67
870 K=S:POKE254,K/256:POKE253,K-PEEK(254)
    *256:POKE780,253              :rem   14
880 K=E+1:POKE782,K/256:POKE251,K-PEEK(78
    2)*256:POKE56333,127:SYS6000:rem  203
890 POKE56333,129                 :rem  156
900 REM                           :rem  125
910 PRINT"{GIU'}FATTO.{GIU'}":GOTO290
                                  :rem  199
920 REM                           :rem  127
930 REM                           :rem  128
940 PRINT"{CLR}{RVS}*** LOAD ***
    { 2 GIU'}"                    :rem  209
950 PRINT"{ 2 GIU'}PREMERE {RVS}RETURN
    {OFF} PER USCIRE DAL LOAD{GIU'}"
                                  :rem   89
960 F$="":INPUT"{ 2 GIU'}NOME DEL FILE";F
    $:IFF$=""THENPRINT:GOTO290  :rem  129
970 PRINT:PRINT"{ 2 GIU'}{RVS}N{OFF}ASTRO
     O {RVS}D{OFF}ISCO: (N/D)"   :rem  134
980 GETA$:IFA$<>"N"ANDA$<>"D"THEN980
                                  :rem   42
990 DV=1-7*(A$="D"):IFDV=8THENF$="0:"+F$
                                  :rem  164
1000 T$=F$:ZK=PEEK(53)+256*PEEK(54)-LEN(T
     $):POKE782,ZK/256            :rem   39
1010 POKE781,ZK-PEEK(782)*256:POKE780,LEN
     (T$):SYS65469                :rem  144
1020 POKE780,1:POKE781,DV:POKE782,1:SYS65
     466                          :rem  104
1030 POKE780,0:SYS65493           :rem   50
1040 REM                          :rem  169
1050 PRINT"{GIU'}FATTO.":GOTO290:rem  226
1060 REM                          :rem  171
1070 REM                          :rem  172
1080 REM CICALINO                 :rem  239
1090 POKE54296,15:POKE54277,45:POKE54278,
     165                          :rem  215
1100 POKE54276,33:POKE54273,6:POKE54272,5
                                  :rem   41
1110 FORT=1TO200:NEXT:POKE54276,32:POKE54
     273,0:POKE54272,0:RETURN    :rem  201
1120 REM CAMPANELLO               :rem  132
1130 POKE54296,15:POKE54277,0:POKE54278,2
     47                           :rem  154
1140 POKE54276,17:POKE54273,40:POKE54272,
     0                            :rem   88
1150 FORT=1TO100:NEXT:POKE54276,16:RETURN
                                  :rem   59
1160 PRINTC$;"{RVS} NON IN PAGINA ZERO O
     SUL KERNAL":GOTO1080        :rem  251
1170 X1=I+K:X2=INT(X1/256):X3=X1-256*X2
                                  :rem  233
1180 POKE6020,X3:POKE6021,X2:POKE56333,12
     7                            :rem  147
1190 SYS6015:POKE56333,129:RETURN
                                  :rem  229
```

Ultrawedge: prima parte

Prima parte
(la seconda parte verrà pubblicata sul prossimo numero).

```
40960 :085,186,142,062,207,160,074
40966 :003,140,112,207,136,032,124
40972 :095,198,173,103,207,208,228
40978 :008,173,104,207,208,003,209
40984 :076,193,195,160,001,177,058
40990 :043,208,005,169,023,076,042
40996 :119,193,169,000,141,253,143
41002 :207,141,254,207,165,043,035
41008 :133,089,165,044,133,090,190
41014 :160,001,177,089,240,019,228
41020 :072,136,177,089,133,089,244
41026 :104,133,090,238,253,207,067
41032 :208,236,238,254,207,208,143
41038 :231,165,087,141,251,207,136
41044 :165,088,141,252,207,206,119
41050 :253,207,208,008,173,254,169
41056 :207,240,027,206,254,207,213
41062 :173,103,207,024,109,251,201
41068 :207,141,251,207,173,104,167
41074 :207,109,252,207,141,252,002
41080 :207,144,222,076,066,195,006
41086 :165,087,141,114,207,165,237
41092 :088,141,115,207,173,103,191
41098 :207,141,118,207,173,104,064
41104 :207,141,119,207,160,000,210
41110 :140,255,199,185,244,162,055
41116 :240,006,032,210,255,200,075
41122 :208,245,165,043,133,087,019
41128 :165,044,133,088,160,003,249
41134 :200,177,087,208,024,200,046
41140 :200,177,087,208,003,076,163
41146 :163,162,160,001,177,087,168
41152 :072,136,177,087,133,087,116
41158 :104,133,088,208,225,201,133
41164 :203,208,026,140,252,199,208
41170 :200,177,087,201,032,208,091
41176 :011,200,177,087,201,032,156
41182 :240,249,201,164,240,037,073
41188 :172,252,199,208,025,201,005
41194 :137,240,028,201,141,240,197
41200 :024,201,167,240,020,201,069
41206 :138,240,016,201,044,208,069
41212 :005,173,255,199,208,007,075
41218 :169,000,141,255,199,240,238
41224 :165,200,165,088,133,092,083
41230 :152,024,101,087,144,002,012
41236 :230,092,133,091,169,032,255
41242 :162,014,157,000,002,202,051
41248 :016,250,162,000,177,087,212
41254 :240,021,201,032,208,004,232
41260 :200,232,208,244,201,058,163
41266 :176,009,201,048,144,005,121
41272 :157,000,002,176,239,142,004
41278 :254,199,162,014,189,000,112
41284 :002,201,032,208,011,202,212
41290 :016,246,169,000,141,255,133
41296 :199,076,175,160,140,255,061
41302 :199,140,252,199,160,004,016
41308 :189,000,002,201,032,240,244
41314 :009,153,105,207,136,048,244
41320 :011,202,016,240,169,048,022
41326 :153,105,207,136,016,250,209
```

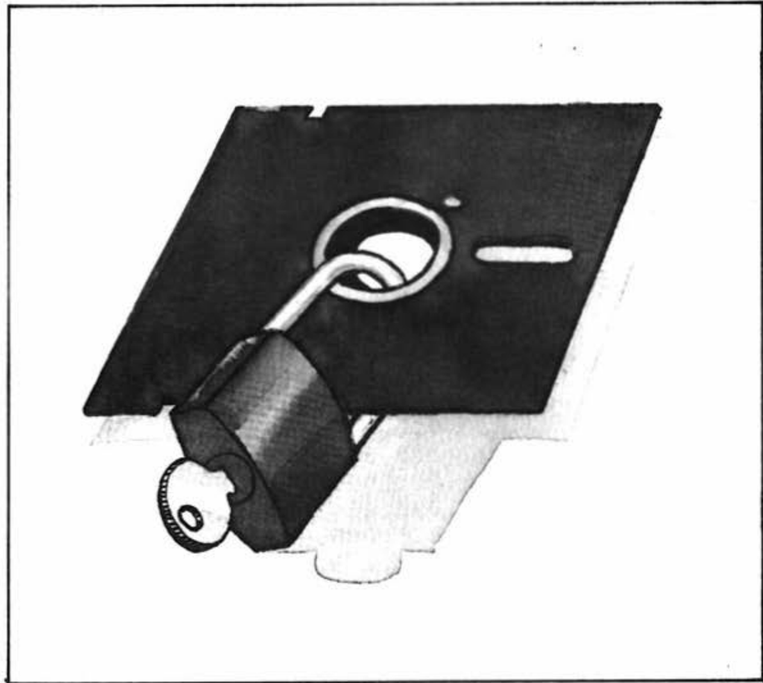
```
41332 :032,011,195,173,114,207,080
41338 :141,253,207,173,115,207,194
41344 :141,254,207,165,043,133,047
41350 :089,165,044,133,090,160,047
41356 :002,177,089,205,103,207,155
41362 :208,008,200,177,089,205,009
41368 :104,207,240,047,160,000,142
41374 :177,089,141,255,207,200,203
41380 :177,089,208,006,172,252,044
41386 :199,076,175,160,133,090,235
41392 :173,255,207,133,089,173,182
41398 :118,207,024,109,253,207,076
41404 :141,253,207,173,119,207,008
41410 :109,254,207,141,254,207,086
41416 :076,139,161,173,253,207,185
41422 :141,103,207,173,254,207,011
41428 :141,104,207,032,221,194,087
41434 :162,255,232,189,105,207,088
41440 :201,048,240,248,142,253,076
41446 :199,169,005,056,237,253,125
41452 :199,141,253,199,173,254,175
41458 :199,205,253,199,176,080,074
41464 :056,165,045,229,091,170,236
41470 :165,046,229,092,141,255,158
41476 :207,165,045,133,095,056,193
41482 :233,001,133,093,165,046,169
41488 :133,096,233,000,133,094,193
41494 :160,000,240,019,177,093,199
41500 :145,095,138,208,008,173,027
41506 :255,207,240,017,206,255,190
41512 :207,202,136,208,237,177,183
41518 :093,145,095,198,094,198,101
41524 :096,208,231,032,121,162,134
41530 :238,254,199,238,252,199,158
41536 :230,045,208,172,230,046,227
41542 :208,168,169,005,056,237,145
41548 :253,199,170,160,000,189,023
41554 :105,207,145,091,232,200,038
41560 :224,005,208,245,169,046,217
41566 :032,210,255,173,254,199,193
41572 :056,237,253,199,240,009,070
41578 :170,169,032,145,091,200,145
41584 :202,208,250,172,252,199,115
41590 :076,175,160,165,087,133,146
41596 :095,165,088,133,096,160,093
41602 :001,177,095,141,255,207,238
41608 :136,177,095,024,105,001,162
41614 :145,095,072,169,000,109,220
41620 :255,207,200,145,095,133,159
41626 :096,104,133,095,177,095,086
41632 :208,227,096,165,043,133,008
41638 :087,165,044,133,088,173,088
41644 :114,207,141,251,207,173,241
41650 :115,207,141,252,207,160,236
41656 :001,177,087,208,003,076,224
41662 :142,195,136,177,087,133,036
41668 :089,200,177,087,133,090,204
41674 :200,173,251,207,145,087,241
41680 :200,173,252,207,145,087,248
41686 :173,251,207,024,109,118,072
41692 :207,141,251,207,173,119,038
41698 :207,109,252,207,141,252,114
41704 :207,165,089,133,087,165,054
41710 :090,133,088,076,183,162,202
41716 :073,078,032,067,079,082,143
41722 :083,079,000,056,032,240,228
41728 :255,202,024,032,240,255,240
41734 :160,039,169,032,145,209,248
41740 :136,016,251,173,024,208,052
41746 :041,002,240,002,169,007,223
41752 :168,169,004,170,032,186,241
41758 :255,169,000,032,189,255,162
41764 :032,192,255,176,054,162,139
41770 :004,032,201,255,176,047,245
41776 :173,136,002,133,088,169,237
41782 :000,133,087,162,025,160,109
41788 :000,177,087,032,103,163,110
41794 :032,210,255,165,144,208,056
41800 :022,200,192,040,144,239,141
41806 :032,142,195,169,040,024,168
41812 :101,087,144,002,230,088,224
41818 :133,087,202,208,220,169,085
41824 :004,032,195,255,076,204,094
41830 :255,041,127,201,032,176,166
41836 :004,009,064,208,015,201,097
41842 :064,144,011,201,096,176,038
41848 :004,009,032,208,003,024,144
41854 :105,064,096,173,051,166,013
41860 :133,093,173,052,166,133,114
41866 :094,160,000,076,133,193,026
41872 :147,013,032,032,032,032,176
41878 :032,069,076,069,078,067,029
41884 :079,032,067,079,077,065,043
41890 :078,068,073,032,068,073,042
41896 :032,085,076,084,082,065,080
41902 :087,069,068,071,069,032,058
41908 :032,032,032,032,013,013,078
```

```
41914 :042,032,032,083,084,065,012
41920 :084,079,032,068,073,083,099
41926 :067,079,032,032,032,032,216
41932 :032,032,042,088,032,083,001
41938 :084,065,077,080,065,032,101
41944 :086,073,068,069,079,013,092
41950 :042,065,032,065,085,084,083
41956 :079,032,078,085,077,047,114
41962 :082,069,047,078,085,077,160
41968 :032,032,042,033,032,067,222
41974 :079,076,068,032,083,084,156
41980 :065,082,084,032,032,032,067
41986 :032,013,042,066,032,083,014
41992 :067,082,047,076,071,032,127
41998 :066,076,079,067,067,072,185
42004 :073,032,032,032,042,035,010
42010 :032,087,069,068,071,069,166
42016 :032,083,085,076,032,057,141
42022 :013,042,067,032,067,079,082
42028 :080,073,065,032,070,073,181
42034 :076,069,047,068,073,083,210
42040 :067,079,032,042,061,032,113
42046 :078,085,079,086,079,032,245
42052 :035,032,068,082,073,086,188
42058 :069,032,032,013,042,068,074
42064 :032,068,073,082,069,067,215
42070 :084,079,082,089,032,032,228
42076 :032,032,032,032,032,032,028
42082 :042,043,032,067,079,076,181
42088 :079,082,069,032,066,079,255
42094 :082,068,079,032,032,032,179
42100 :032,013,042,070,032,077,126
42106 :069,077,079,082,073,065,055
42112 :032,076,073,066,069,082,014
42118 :065,032,032,032,042,045,126
42124 :032,067,079,076,079,082,043
42130 :069,032,083,070,079,078,045
42136 :068,079,032,013,042,073,203
42142 :032,073,078,073,090,073,065
42148 :065,076,073,090,090,065,111
42154 :032,068,082,073,086,069,068
42160 :042,092,032,067,079,076,052
42166 :079,082,069,032,084,069,085
42172 :083,084,079,032,032,013,255
42178 :042,076,032,073,078,068,051
42184 :046,032,073,078,073,090,080
42190 :073,065,076,069,032,032,041
42196 :032,032,042,063,032,086,243
42202 :073,083,085,065,076,073,161
42208 :090,090,065,032,077,069,135
42214 :078,085,032,013,042,077,045
42220 :032,069,078,084,082,065,134
42226 :032,073,078,032,077,079,101
42232 :078,073,084,079,082,032,164
42238 :042,047,032,076,079,065,083
42244 :068,032,080,082,071,046,127
42250 :032,032,032,013,042,078,239
42256 :032,070,079,082,077,065,165
42262 :084,084,065,032,068,073,172
42268 :083,067,079,032,032,032,097
42274 :042,044,032,076,079,065,116
42280 :068,032,080,082,071,046,163
42286 :032,076,047,077,032,032,086
42292 :013,042,080,032,083,084,130
42298 :065,077,080,065,032,070,191
42304 :073,076,069,032,083,069,210
42310 :081,046,032,042,094,032,141
42316 :076,079,065,068,043,082,233
42322 :085,078,032,080,082,071,254
42328 :046,032,032,032,013,042,029
42334 :081,032,069,083,067,069,239
42340 :032,087,069,068,071,069,240
42346 :032,032,032,032,032,032,042
42352 :032,042,095,032,083,065,205
42358 :086,069,032,080,082,071,026
42364 :046,032,032,032,013,042,065
42370 :082,032,082,069,078,065,026
42376 :077,069,032,032,032,032,154
42382 :032,032,032,032,032,032,078
42388 :032,042,058,032,076,079,211
42394 :065,068,032,070,073,076,026
42400 :069,032,083,069,081,013,251
42406 :042,083,032,083,067,082,043
42412 :065,084,067,072,032,032,012
42418 :032,032,032,032,032,032,114
42424 :032,032,042,059,032,083,208
42430 :065,086,069,032,070,073,073
42436 :076,069,032,083,069,081,094
42442 :013,042,085,032,078,079,019
42448 :045,078,069,087,032,032,039
42454 :032,032,032,032,032,032,150
42460 :032,032,032,042,064,032,198
42466 :076,079,065,068,032,070,104
42472 :073,076,069,032,085,083,138
42478 :082,013,042,086,032,086,067
42484 :065,076,073,068,065,084,163
42490 :069,032,032,032,032,032,223
42496 :032,032,032,032,042,042,212
42502 :032,083,065,086,069,032,117
42508 :070,073,076,069,032,085,161
42514 :083,082,013,013,032,032,017
42520 :032,032,032,032,032,032,216
42526 :032,032,032,032,066,089,057
42532 :032,074,073,077,032,075,143
42538 :076,073,084,090,073,078,004
42544 :071,013,000,144,163,000,183
42550 :000,000,255,255,000,000,052
```

File Protector

di **J. Dearinger**
trad. ed adatt.
di **L. Priotto**

Con questo programma non dovrete più preoccuparvi di cancellare accidentalmente qualche file importante dai vostri dischi. Esso infatti vi consente di proteggere dalla cancellazione o dalla sovrascrittura sia singoli file che l'intero disco.



Vi è mai capitato di cancellare un certo programma dal disco e poi immediatamente accorgervi di aver agito sul programma sbagliato? Può essere stato a causa di una vostra distrazione, oppure potreste aver usato l'asterisco, ed aver cancellato magari quattro o cinque file il cui nome inizia con gli stessi caratteri...

Alcuni computer, come ad esempio Apple ed Atari, permettono di "aprire" e "chiudere" i file su disco, offrendo quindi una valida protezione contro errori nelle cancellazioni. Non si confondano i termini "aprire" e "chiudere" qui usati con quelli relativi alle istruzioni OPEN e CLOSE: in questo articolo intenderemo sempre per "chiusura" di un file la sua protezione software contro eventuali cancellazioni o sovrascritture, e per "apertura" di un file il procedimento inverso.

I computer Commodore non posseggono comandi atti a proteggere dei file su disco da cancellazioni accidentali, ma il drive 1541 possiede all'interno del DOS (Disk Operating System) delle routine in linguaggio macchina che possono essere usate per questo scopo.

Ogni volta che il 1541 inizia la procedura di cancellazione di un file per prima cosa cerca il file all'interno della directory, per assicurarsi della sua esistenza. Una volta trovato, il drive acquisisce una serie di informazioni sul file stesso, informazioni che sono contenute con il nome del file nella directory alla traccia 18. Queste informazioni sono tra l'altro la traccia e il settore del disco in cui inizia il file vero e proprio, la lunghezza del file e il suo tipo (PRG, SEQ, ecc.).

Un'altra routine del DOS controlla il primo byte relativo ad ogni file che normalmente è un numero compreso tra 128 e 132. Se il bit 6 di questo byte è posto a 1, il DOS tratta il file come "chiuso", e non lo cancellerà quindi dal disco.

Il programma qui presentato usa i comandi di accesso diretto al disco per chiudere i file, in modo che essi non possano più essere cancellati, a meno che, ovviamente, essi vengano riaperti.

Quando un file è stato chiuso, in corrispondenza ad esso nella directory compare un segno di minore (<), per ricordarvi il suo stato "indelebile".

L'opzione di cancellazione di un file è l'unica che permette le "wild card" all'interno del nome del file – ricordiamo che le wild card sono asterisco e punto di domanda –, mentre tutte le altre opzioni del menu richiedono la digitazione dell'intero nome del file.

Il programma fornisce anche l'opzione di "chiusura" dell'intero disco: essa blocca qualunque operazione che comporti l'accesso al disco per operazioni

diverse dalla lettura, fino a che il disco stesso venga "riaperto" attraverso opzione.

Alcune raccomandazioni

Come è ovvio, i file protetti in questo modo contro la cancellazione verranno comunque cancellati da un'operazione di formattazione disco con

OPEN15,8,15,"NO:NOMEDISCO,ID"

o da una operazione di azzeramento della directory con:

OPEN15,8,15,"NO:NOMEDISCO"

In particolare quest'ultimo comando non avrà effetto solo se è stato "chiuso" l'intero disco.

Infine si raccomanda di non usare questo programma con dischi contenenti programmi commerciali protetti: questi ultimi infatti usano spesso la traccia 18 per le protezioni anticopia e, dal momento che "File Protector" ne modifica alcuni byte, può essere che non si riesca più a caricare i programmi per l'intervento delle protezioni.

Il sistema usato per "chiudere" l'intero disco è simile a quello usato per i singoli file: nella traccia 18, settore 0, è contenuta la BAM (Block Availability Map. I suoi primi due byte dicono al 1541 dove può essere trovato il primo blocco della directory, contenente i primi otto nomi di file. Il terzo byte della BAM indica con quale drive è stato formattato il dischetto: un valore 65 indica un 1541 con formato 4040. Se il valore è diverso, ciò indica che il disco è stato formattato su un altro tipo di drive, e il DOS leggerà il disco, rifiutandosi invece di scrivervi sopra.

Scrivendo quindi un numero diverso all'interno di questo byte, il disco può essere protetto contro la scrittura. Il programma modifica inoltre la locazione 166 da 65 in 66, per indicare visivamente se un disco è stato o meno protetto contro la scrittura. La testata della directory di un disco "chiuso" conterrà infatti il codice "2B", invece del familiare "2A", a fianco del codice identificatore del disco.

File Protector

```
100 PRINT"{CLR}":Q$=CHR$(13)        :rem 151
110 U$="{RVS}{ 40 SPAZI}":KB=198:SF=653:K
    L=203:NS=12                     :rem 16
120 F1=4:F7=3:POKE53280,6:POKE53281,12
                                    :rem 106
130 GOSUB1830:GOTO1630              :rem 29
140 PRINT#15,"U1";2;0;T;S:GOTO1010
                                    :rem 203
150 PRINT#15,"B-P 2 0":PRINT#15,"U2";2;0;
    T;S:GOTO1010                    :rem 174
160 T=PEEK(681):S=PEEK(682):RETURN
                                    :rem 55
170 :                               :rem 210
180 REM ********************** :rem 67
190 REM * CHIUDE TUTTI I FILE * :rem 135
200 REM ********************** :rem 60
210 :                               :rem 205
220 PRINT"{CLR}"SPC(NS)"CHIUDI TUTTI I FI
    LES":PRINTU$:GOSUB1000:T=18:S=1
                                    :rem 10
230 P=2:GOSUB140                    :rem 163
240 GET#2,A$:T1=ASC(A$+CHR$(0)):GET#2,A$:
    S1=ASC(A$+CHR$(0))              :rem 199
250 FORI=0TO7:PRINT#15,"B-P";2;P+32*I
                                    :rem 136
260 GET#2,A$:A=ASC(A$+CHR$(0)):IFA=.THEN2
    90                              :rem 143
270 IFAAND64THEN290                 :rem 112
280 PRINT#15,"B-P";2;P+32*I:PRINT#2,CHR$(
    AOR64);                         :rem 251
290 NEXT:GOSUB150:IFETHENRETURN :rem 71
300 PRINT"TRACCIA"T"SETTORE"S"CHIUSO":T=T
    1:S=S1:IFTTHEN230               :rem 250
310 GOSUB1070:GOSUB1080:RETURN      :rem 121
320 :                               :rem 207
330 REM ********************** :rem 22
340 REM * LEGGE LA DIRECTORY *  :rem 117
350 REM ********************** :rem 24
360 :                               :rem 211
370 H$=CHR$(18):PRINT"{CLR}{RVS}PREMI [SH
    IFT] PER PAUSA{GIU'}"           :rem 177
380 GOSUB1000:PRINT#15,"M-R"CHR$(144)CHR$
    (7)CHR$(23)                     :rem 147
390 FORI=0TO22:GET#15,A$:H$=H$+(A$+CHR$(0
    )):NEXT:PRINTH$"{BLK}":POKE140,1:T=18
    :S=1                            :rem 91
400 GOSUB140:SYS828:GOSUB160:IFPEEK(SF)TH
    ENWAITSF,1,1                    :rem 100
410 IFTTHEN400                      :rem 59
420 PRINT#15,"M-R"CHR$(250)CHR$(2):GET#15
    ,LO$:PRINT#15,"M-R"CHR$(252)CHR$(2)
                                    :rem 222
430 GET#15,HI$:PRINTASC(LO$+CHR$(0))+256*
    ASC(HI$+CHR$(0))" BLOCCHI LIBERI"
                                    :rem 167
440 GOSUB1070:GOSUB1080:POKE140,0:RETURN
                                    :rem 215
450 :                               :rem 211
460 REM ***************************
                                    :rem 236
470 REM * INPUT DEL NOME DEL FILE *
                                    :rem 92
480 REM ***************************
                                    :rem 238
490 :                               :rem 215
500 PRINT"{GIU'}[F1] ESCI{ 13 SPAZI}[F7]
    DIRECTORY                       :rem 146
510 PRINT"{GIU'}NOME FILE? "CU$;:POKEKB,0
    :F$=""                          :rem 116
520 KQ=PEEK(KL):GETA$:IFA$=""ANDKQ=64THEN
    520                             :rem 135
530 IFKQ=F1ORKQ=F7THENPRINTCHR$(20):RETUR
    N                               :rem 248
540 IFA$=CHR$(20)ANDF$=""THEN520:rem 2
550 IFA$=CHR$(13)ANDF$<>""THENPRINTCHR$(2
    0):RETURN                       :rem 6
560 IFA$=CHR$(13)ANDF$=""THEN520:rem 6
570 IFA$=CHR$(20)THENPRINTCHR$(20)A$CU$;:
    F$=LEFT$(F$,LEN(F$)-1):GOTO520
                                    :rem 102
580 PRINTCHR$(20)A$CU$;:F$=F$+A$:GOTO520
                                    :rem 31
590 :                               :rem 216
600 REM *****************           :rem 110
610 REM * CHIUDE UN FILE *          :rem 68
620 REM *****************           :rem 112
630 :                               :rem 211
640 PRINT"{CLR}";                   :rem 57
650 PRINTSPC(NS)"CHIUDI UN FILE":PRINTU$
                                    :rem 253
660 GOSUB500:IFKQ=F1THENRETURN      :rem 217
670 IFKQ=F7THENGOSUB370:GOTO650 :rem 217
680 POKE679,1:GOSUB1160:GOSUB1000:T=18:S=
    1                               :rem 252
690 GOSUB140:SYS828:A=PEEK(252):IFATHEN72
    0                               :rem 40
```

```
700 GOSUB160:IFTTHEN690                :rem 153
710 PRINT"{GIU'}{RED}FILE NON TROVATO
    {BLK}{GIU'}":GOTO660               :rem 128
720 IF(AAND64)THENPRINTF$:PRINT" E' GIA'
    CHIUSO":GOSUB1070:GOSUB1080:GOTO760
                                       :rem 108
730 P=PEEK(255):GOSUB140:PRINT#15,"B-P";2
    ;P:PRINT#2,CHR$(AOR64);            :rem 32
740 GOSUB150:IFETHEN640                :rem 136
750 GOSUB1070:PRINTF$:PRINT"E' CHIUSO":GO
    SUB1080                            :rem 218
760 POKE679,0:GOTO640                  :rem 219
770 :                                  :rem 216
780 REM ****************               :rem 35
790 REM * APRE UN FILE *               :rem 195
800 REM ****************               :rem 28
810 :                                  :rem 211
820 PRINT"{CLR}";                      :rem 57
830 PRINTSPC(NS)"APRI UN FILE":PRINTU$
                                       :rem 115
840 GOSUB500:IFKQ=F1THENRETURN         :rem 217
850 IFKQ=F7THENGOSUB370:GOTO830        :rem 217
860 POKE679,1:GOSUB1160:GOSUB1000:T=18:S=
    1                                  :rem 252
870 GOSUB140:SYS828:A=PEEK(252):IFATHEN90
    0                                  :rem 40
880 GOSUB160:IFTTHEN870                :rem 162
890 PRINT"{GIU'}{RED}FILE NON TROVATO
    {BLK}{GIU'}":GOTO840               :rem 137
900 IF(AAND64)=.THENPRINTF$Q$"E' GIA' APE
    RTO":GOSUB1070:GOSUB1080:GOTO940
                                       :rem 133
910 P=PEEK(255):GOSUB140:PRINT#15,"B-P";2
    ;P:PRINT#2,CHR$(AAND135);          :rem 129
920 GOSUB150:IFETHEN820                :rem 136
930 GOSUB1070:PRINTF$:PRINT"E' APERTO":GO
    SUB1080                            :rem 218
940 POKE679,0:GOTO820                  :rem 219
950 :                                  :rem 216
960 REM *********************************
    ******                             :rem 233
970 REM * LEGGE CANALE DEGLI ERRORI DEL D
    RIVE *                             :rem 103
980 REM *********************************
    ******                             :rem 235
990 :                                  :rem 220
1000 CLOSE2:CLOSE15:OPEN15,8,15,"I0":OPEN
     2,8,2,"#"                         :rem 35
1010 INPUT#15,E,EM$,ET,ES              :rem 183
1020 IFE=73ORE=26THENGOSUB1070:GOTO1050
                                       :rem 121
1030 IFETHENPRINT"{RED}{RVS}ERRORE:":PRIN
     TE","EM$","ET","ES"{BLK}":GOSUB1070:
     END                               :rem 121
1040 RETURN                            :rem 165
1050 IFE=73THENPRINT"IL DISCO E' CHIUSO":
     GOSUB1080:RETURN                  :rem 28
1060 PRINT"TOGLI L'ADESIVO DAL DISCO":GOS
     UB1080:RETURN                     :rem 117
1070 CLOSE2:CLOSE15:RETURN             :rem 160
1080 PRINT"{RVS}{GIU'}PREMI UN TASTO":POK
     EKB,0                             :rem 186
1090 KQ=PEEK(KL):GETA$:IFA$=""ANDKQ=64THE
     N1090                             :rem 237
1100 RETURN                            :rem 162
1110 :                                 :rem 253
1120 REM ********************************
     **********                        :rem 140
1130 REM * MEMORIZZA NOME PER LINGUAGGIO
     MACCHINA *                        :rem 235
1140 REM ********************************
     **********                        :rem 142
1150 :                                 :rem 1
1160 IFLEN(F$)<16THENF$=F$+CHR$(160):GOTO
     1160                              :rem 35
1170 FORI=1TOLEN(F$):POKE683+I,ASC(MID$(F
     $,I,1)):NEXT:POKE700,0:RETURN
                                       :rem 240
1180 :                                 :rem 4
1190 REM **************************
                                       :rem 201
1200 REM * CHIUDE L'INTERO DISCO *
                                       :rem 99
1210 REM **************************
                                       :rem 194
1220 :                                 :rem 255
1230 PRINT"{CLR}"SPC(NS)"CHIUDI L'INTERO
     DISCO":PRINTU$                    :rem 242
1240 PRINT"{GIU'}INSERISCI DISCO NEL DRIV
     E{ 2 SPAZI}[F1] ESCE":FORTD=1TO900:N
     EXT:GOSUB1080                     :rem 203
1250 IFKQ=F1THENRETURN                 :rem 182
1260 GOSUB1000:T=18:S=0:GOSUB140:PRINT#15
     ,"B-P 2 2"                        :rem 106
1270 PRINT#2,CHR$(66);:PRINT#15,"B-P 2 16
     6":PRINT#2,CHR$(66);:GOSUB150:IFETHE
     N1230                             :rem 179
1280 GOSUB1070:GOSUB1000:GOSUB1070
                                       :rem 16
1290 PRINT"IL DISCO E' ORA PROTETTO DA SC
     RITTURA":GOSUB1080:RETURN         :rem 86
1300 :                                 :rem 254
1310 REM ********************          :rem 241
1320 REM * CANCELLA UN FILE *          :rem 244
1330 REM ********************          :rem 243
1340 :                                 :rem 2
1350 PRINT"{CLR}";                     :rem 104
1360 PRINTSPC(NS)"CANCELLA UN FILE":PRINT
     U$                                :rem 169
1370 GOSUB500:IFKQ=F1THENRETURN        :rem 8
1380 IFKQ=F7THENGOSUB370:GOTO1360
                                       :rem 55
1390 INPUT"{GIU'}SEI SICURO";A$:IFA$<>"S"
     THEN1350                          :rem 63
1400 GOSUB1000:PRINT#15,"S0:"+F$:INPUT#15
     ,E,EM$,ET,ES:IFE>1THENGOSUB1020:GOTO
     1350                              :rem 114
1410 GOSUB1070:IFET=.THENPRINT"{RED}IL FI
     LE E' CHIUSO O NON SUL DISCO{BLK}":G
     OSUB1080:GOTO1350                 :rem 210
1420 IFET>1THENPRINTET;EM$:GOSUB1080:GOTO
     1350                              :rem 99
1430 PRINTF$:PRINT"CANCELLATO":GOSUB1080:
     GOTO1350                          :rem 95
1440 :                                 :rem 3
1450 REM *****************             :rem 120
1460 REM * APRE IL DISCO *             :rem 50
1470 REM *****************             :rem 122
1480 :                                 :rem 7
1490 PRINT"{CLR}"SPC(NS)" APRI DISCO":PRI
     NTU$                              :rem 44
1500 PRINT"{GIU'}INSERISCI DISCO NEL DRIV
     E{ 3 SPAZI}[F1] ESCE":FORTD=1TO900:N
     EXT:GOSUB1080                     :rem 202
1510 IFKQ=F1THENRETURN                 :rem 181
1520 GOSUB1000:T=18:S=0:GOSUB140:rem 134
1530 PRINT#15,"M-W";CHR$(1);CHR$(1);CHR$(
     1);CHR$(65);                      :rem 141
1540 PRINT#15,"B-P 2 2":PRINT#2,CHR$(65);
     :PRINT#15,"B-P 2 166"             :rem 1
```

```
1550 PRINT#2,CHR$(65);:GOSUB150:IFETHEN14
     90                             :rem  44
1560 GOSUB140:GOSUB1070:PRINT"IL DISCO E'
      APERTO"                       :rem 172
1570 GOSUB1080:RETURN               :rem  48
1580 :                              :rem   8
1590 REM ******************         :rem 209
1600 REM * MENU PRINCIPALE *        :rem  27
1610 REM ******************         :rem 202
1620 :                              :rem   3
1630 PRINT"{CLR}{BLK}{GIU'}"SPC(NS-2)"<<P
     ROTETTORE FILE>>"              :rem 118
1640 PRINT"{GIU'} 1) DIRECTORY      :rem 154
1650 PRINT" 2) CHIUDI TUTTI I FILE
                                    :rem 143
1660 PRINT" 3) CHIUDI UN FILE       :rem  81
1670 PRINT" 4) APRI UN FILE         :rem 201
1680 PRINT" 5) CHIUDI L'INTERO DISCO
                                    :rem  72
1690 PRINT" 6) APRI UN DISCO        :rem  31
1700 PRINT" 7) CANCELLA UN FILE     :rem 205
1710 PRINT" 8) ESCI                 :rem 253
1720 PRINT"{ 2 GIU'} INSERISCI LA TUA SCE
     LTA"                           :rem 153
1730 GETA$:IFA$=""THEN1730          :rem 185
1740 C=VAL(A$):IFC<1ORC>8THEN1730
                                    :rem 178
1750 ONCGOSUB370,220,640,820,1230,1490,13
     50,1770                        :rem 239
1760 GOTO1630                       :rem 209
1770 CLOSE2:CLOSE15:END             :rem 158
1780 :                              :rem  10
1790 REM *******************************
     ***                            :rem 115
1800 REM * VARIABILI E LINGUAGGIO MACCHIN
     A *                            :rem 243
1810 REM *******************************
     ***                            :rem 108
1820 :                              :rem   5
1830 POKE679,0:POKE140,0:CU$=CHR$(31)+CHR
     $(161)+CHR$(144)               :rem 107
1840 FORQZ=828TO1006:READZQ:POKEQZ,ZQ:NEX
     T:RETURN                       :rem  39
1850 DATA 169,0,141,168,2,133,253,162,2,3
     2,198,255,32                   :rem  51
1860 DATA 228,255,141,169,2,32,228,255,14
     1,170,2,230,253                :rem 197
1870 DATA 230,253,32,228,255,133,252,165,
     253,133,255,32,228             :rem  96
1880 DATA 255,32,228,255,230,253,230,253,
     160,0,32,228,255               :rem 246
1890 DATA 230,253,153,189,2,200,24,192,16
     ,144,242,160,243               :rem 244
1900 DATA 32,228,255,230,253,165,144,240,
     3,141,168,2,200                :rem 183
1910 DATA 208,241,165,252,208,7,173,168,2
     ,208,22,240,191                :rem 199
1920 DATA 165,140,240,3,32,199,3,173,167,
     2,240,3,32                     :rem 196
1930 DATA 175,3,173,168,2,240,171,32,204,
     255,96,160,0                   :rem  47
1940 DATA 185,172,2,240,8,217,189,2,208,7
     ,200,208,243                   :rem  52
1950 DATA 238,168,2,96,169,0,133,252
                                    :rem 250
1960 DATA 96,160,0,185,189,2,240,6,32,210
     ,255,200,208                   :rem  46
1970 DATA 245,165,252,41,64,240,10,169,14
     4,32,210,255,169,60,32,210,255,169
                                    :rem 116
1980 DATA 13,32,210,255,169,144,32,210,25
     5,96                           :rem 172
```

ERRATA CORRIGE

Alcuni lettori ci hanno segnalato due tipi di problemi nell'utilizzo di "Relfiling System 1.0": il primo riguarda l'apparizione di un messaggio d'errore "File not open" quando si cerca di usare l'opzione 3 per l'inserimento dei dati.

Ciò non è dovuto ad un errore nel programma, ma ad una errata spiegazione - nell'articolo - della procedura da seguire. Dopo aver utilizzato l'opzione 0 per creare il file su disco, occorre infatti - prima di effettuare qualsiasi operazione di lettura o scrittura sul file - selezionare l'opzione 1 di definizione del file corrente.

Una volta definito quest'ultimo, si potranno utilizzare a piacere tutte le opzioni del menu di selezione.

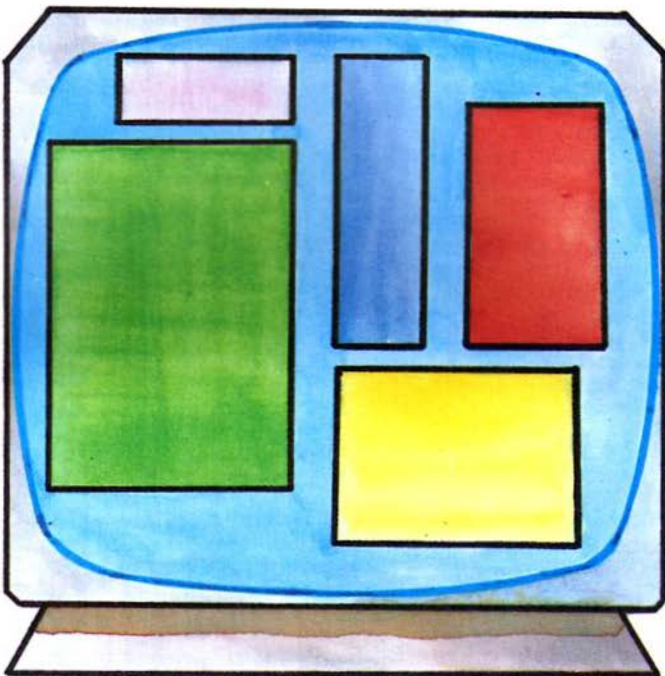
Il secondo inconveniente consiste in un messaggio d'errore "Bad subscript", che si verifica quando si cerca di inserire un record il cui numero è superiore al numero totale di record assegnato inizialmente. Anche questo inconveniente non dipende dal programma, ma da un'errata spiegazione nell'articolo: il numero totale di record che si assegna attraverso l'opzione 0 rappresenta infatti il limite massimo delle dimensioni del file, e non può mai essere superato. Quando si crea il file dati con l'opzione 0, conviene quindi fornire al computer un numero totale di record il più elevato possibile in modo che esso possa difficilmente venire superato.
Ci scusiamo con tutti i lettori per l'inconveniente.



Rettangoli

di **D. Fish**
trad. ed adatt.
di **S. Colombo**

Cercate di insegnare bene al-l'"intelligenza artificiale" del vostro serpente: si ricorderà di ogni mossa che fate per conquistare la scacchiera con i vostri rettangoli. Un gioco di strategia che ammette da uno a quattro giocatori.



Dopo aver dato il RUN al programma, per ognuno dei quattro serpenti vi sarà chiesto se è controllato da un giocatore oppure dal computer: le mosse dei serpenti controllati dai giocatori vengono inserite da tastiera, mentre quelle dei serpenti controllati dal computer sono pseudo-casuali.

Potete muovere il vostro serpente nelle quattro direzioni attraverso i tasti I, M, J e K; quando lo muovete da un punto all'altro della scacchiera esso lascia una traccia del suo colore. Con ogni mossa che effettuate il vostro serpente "impara" la vostra strategia: per esempio, supponiamo che esistano tracce lasciate da serpenti avversari sia a sinistra che in basso rispetto al vostro serpente, e voi muovete verso l'alto. Da questo momento in poi quando il vostro serpente si troverà in una situazione analoga muoverà da solo verso l'alto.

Se il serpente incontra una configurazione che non ha ancora "imparato", vi chiederà una direzione: quella che gli darete verrà quindi riutilizzata dal serpente quando si troverà in circostanze analoghe.

Un serpente si può intrappolare, se gli date istruzioni che formino un ciclo chiuso con una precedente istruzione: se ad esempio gli dite di andare a destra, e andando a destra trova una configurazione in cui precedentemente l'avevate istruito per andare a sinistra, rimarrà "intrappolato" da queste due istruzioni.

Un serpente intrappolato può darsi venga liberato, nello svolgersi del gioco, da un altro serpente che modifichi la configurazione circostante.

I rettangoli vengono conquistati compiendo un percorso chiuso attraverso i quattro punti costituenti appunto i vertici di un rettangolo: in questo caso lo stesso viene colorato con il vostro colore e guadagnate un punto.

Il gioco finisce quando tutti i rettangoli sono colorati, oppure tutti i serpenti risultano intrappolati; il giocatore che avrà colorato il maggior numero di rettangoli vince la partita.

All'inizio non sarà facile ricordare quale strategia abbiamo "insegnato" al nostro serpente attraverso le mosse precedenti, ma con la pratica di questo gioco molto divertente potrete imparare a programmare al meglio questa piccola "intelligenza artificiale".

Rettangoli

```
10 POKE53281,0:POKE53280,0:PRINT"{WHT}"
                                  :rem 198
20 DIMIN(15,4):FORA=984TO1023:POKEA,32:NE
   XT                              :rem  75
30 DR(0)=-40:DR(1)=1:DR(2)=40:DR(3)=-1
                                   :rem 235
```

segue a pag. 51

BANK STREET WRITER

Prodotto da: Broderbund software
Supporto: disco

Presentare un word processor è una delle cose più piacevoli che possa capitare a un giornalista che si occupa di informatica. Dopo le prime schermaglie iniziali con il programma, infatti, si entra subito nel vivo del problema, perché il word processor stesso viene utilizzato per descriverlo. Così, man mano che si scrive e si impiegano le varie opzioni si approfondisce la conoscenza del programma stesso.
Siamo abituati ad utilizzare Easy Script, forse il più diffuso word processor per Commodore, e dobbiamo dire che, a parte alcune difficoltà iniziali, ci troviamo bene.
Nonostante questo, ci piace anche controllare le qualità degli altri, pochi, word processor che circolano in Italia. Dunque Bank street writer.
Qualità immediatamente interessante del programma è quella di possedere al suo interno un programma tutorial: un programma vero e proprio, cioè, che consente di conoscere tutte le opzioni offerte da Bank street senza leggersi il manuale in inglese: utile.
Unica pecca è quella di essere un programma a se stante e di non poter essere richiamato, come uno dei tanti help. disponibili altrove, nel corso stesso del lavoro, in caso di dubbio. Quando cioè serve veramente.
Altra osservazione da compiere, abituati o forse viziati da Easy Script, è che manca completamente sullo schermo, e nel corso stesso della scrittura, una indicazione del numero di righe che si sono scritte. Per chi compie un lavoro come il nostro è importante sapere quanto si è scritto, ma certamente questo non serve per tutti coloro, e sono evidentemente la maggioranza, che utilizzano il computer per scrivere altro che articoli. L'unica informazione fornita dal computer a questo proposito è quella relativa al numero di parole scritte: in tutto ce ne stanno 3000.

Il menu

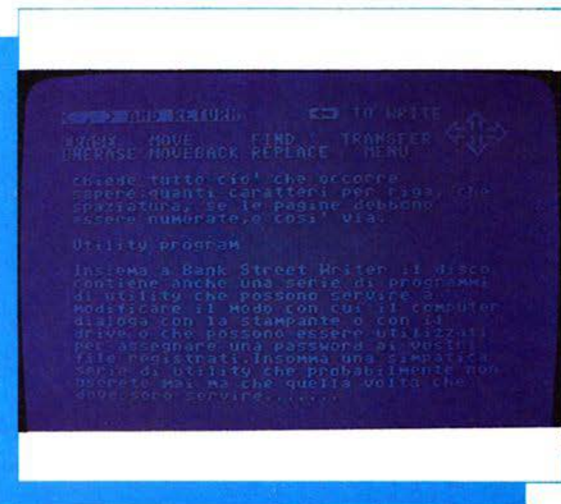
Tutte le funzioni del menu sono elencate in tre schermi successivi nella parte superiore della pagina di testo: un unico tasto serve per scivolare dall'uno all'altro, mentre un altro tasto, space, serve per passare attraverso le varie opzioni.
Questo sistema, proprio mentre lo stiamo sperimentando, ci sembra molto comodo, ed elimina la necessità di ricordarsi tutta una serie di comandi.
Interessanti per altro anche alcune opzioni raggiungibili invece attraverso il tasto CTRL e un altro tasto particolare: marginatura dei paragrafi per esempio, la già ricordata funzione di calcolo delle pagine scritte, la centratura di una riga di testo e così via, secondo una logica molto attenta alla composizione e alla impaginazione del testo vero e proprio.
Oltre alle opzioni che abbiamo descritto, Bank Street Writer offre anche ovviamente quelle stesse che un normale utilizzatore di word processor vuol trovare nel suo programma: move, per spostare il testo, copy, delete, find, replace e così via, secondo una categoria di possibilità ormai ampiamente conosciute.
Il modo per utilizzare queste opzioni, come si è detto, è abbastanza semplice e comunque in genere spiegato chiaramente sul video stesso.
Due parole ancora sulla dimensione del testo sullo schermo e sulla presentazione. Bank Street scrive 38 caratteri per riga e visualizza contemporaneamente sul video 19 righe, che possono naturalmente essere scrollate come più vi piace.

La stampa

Semplicissime francamente anche le operazioni per stampare. Anche in questo caso un menu a domande vi chiede tutto ciò che occorre sapere: quanti caratteri per riga, che spaziatura, se le pagine debbono essere numerate e così via.

Utility program

Insieme a Bank Street Writer, il disco contiene anche una serie di programmi di utility, che possono servire a modificare il modo con cui il computer dialoga con la stampante o con il drive o che possono essere utilizzati per assegnare una password ai vostri file registrati. Insomma, una simpatica serie di utility, che probabilmente non userete mai, ma quella volta che dovessero servire...



SOUND

Rubrica di musica con il C64

Il generatore di suono

La COMMODORE con il C64 è stata la prima ad introdurre di serie la sintesi sonora negli home computer ad un discreto livello qualitativo.
Infatti, esistevano già schede musicali aggiuntive per altre macchine, come ad esempio l'Apple II, schede sicuramente più sofisticate, ma con un prezzo decisamente superiore.
Quello che differenzia principalmente il C64 dai suoi diretti concorrenti è l'essere dotato di un piccolo sintetizzatore analogico indipendente: il chip 6581; questo microprocessore è interamente adibito alla sintesi sonora ed è dotato di innumerevoli caratteristiche.

Il 6581

Il "SID" (Sound Interface Device) è completamente programmabile, anche se, come per la grafica, il BASIC residente non è certamente sufficiente; infatti la programmazione musicale del generatore di suono è realizzata con l'uso delle POKE o con l'uso del linguaggio macchina.
La sintesi sonora avviene per via analogica, ovvero con l'uso di generatori di suono (oscillatori), generatori di inviluppo, filtri, ecc...
Lo stesso principio che regola la maggior parte dei sintetizzatori in circolazione. Il SID è ovviamente compatibile con gli altri chip del C64 e ne sfrutta le capacità di lavoro. Questo espande notevolmente le sue possibilità d'uso, al di là di semplice gadget per videogiochi.

La polifonia

Il SID possiede 3 canali di uscita completamente indipendenti, per cui si può suonare in polifonia a 3 voci oppure in monofonia con uno, due o tre canali contemporaneamente: le voci sono sempre programmabili separatamente.
Nella descrizione specifica il riferimento sarà ovviamente comune ai 3 generatori, anche se più avanti parleremo delle possibili relazioni tra gli oscillatori e delle tecniche di sintesi più avanzate.

Il generatore di suono

Il VCO (Voltage Control Oscillator) è il generatore dell'onda sonora principale e determina la timbrica fondamentale del suono. A disposizione ci sono tre forme d'onda: sawthooth (dente di sega), pulse (onda quadra), triangle (triangolare).
Inoltre è presente un generatore di rumore bianco (noise).
La selezione delle forme d'onda è diretta, ma l'onda PULSE deve essere anche programmata; bisogna infatti determinare la lunghezza del ciclo di pulsazioni e per questo abbiamo a disposizione due parametri: HI e LOW PULSE.
Le differenti timbriche prodotte dalle tre onde sonore permettono la creazione di una vasta gamma di suoni. È così possibile riprodurre suoni reali: fischi, rumori, voci... e strumenti classici come: violini, fiati, organi, ecc.

L'altezza del suono (PITCH) viene determinata dalla frequenza dell'onda sonora prodotta, valore che viene calcolato dal C64 usando due locazioni da inserire con le POKE negli appositi registri del SID. La nota verrà così prodotta con la frequenza voluta.
Nei manuali sono a disposizione i valori per 8 ottave complete; la scala usata è la scala temperata, che è quella con 12 semitoni e dove la nota LA centrale vibra a 440 hertz. È comunque possibile programmare qualsiasi altra scala e determinare l'accordatura.

Il VCA

Il VCA (Voltage Control Amplifier) è l'amplificatore del segnale in uscita dal VCO, ha una gamma di dinamica di 48 db, ed è in pratica il volume regolabile sulla scala (0-15).

L'inviluppo

Il generatore di inviluppo è un elemento importantissimo nella sintesi sonora: è la prima porta verso la manipolazione dell'onda appena creata.
Abbiamo infatti definito la timbrica dell'onda con il PITCH e la WAVEFORM prescelta; dobbiamo ora dare vita all'evento sonoro, definendo i tempi e le forme della sua evoluzione.
L'inviluppo del C64 ha quattro elementi di manipolazione dell'onda: A.D.S.R. ovvero l'ATTACK, il DECAY, il SUSTAIN e il RELEASE.
L'ATTACK è una misura che indica il tempo in cui l'evento sonoro raggiunge il massimo del volume: è programmabile da 2 millesecondi a 8 secondi.
Il DECAY indica il tempo in cui la nota cade dal volume massimo al volume di SUSTAIN: varia da 6 millesecondi a 24 secondi.
Il SUSTAIN è il valore del volume che assume la nota fino al rilascio: varia dal volume 0 al volume massimo.
Il RELEASE è la misura del tempo con cui la nota cade dal volume di SUSTAIN al volume 0: varia da 6 millesecondi a 24 secondi.
L'inviluppo forma il corpo dell'onda, che ora è ben definita, ma non ancora perfezionata: i prossimi passi sono ora i più importanti nella costruzione dell'evento sonoro.

I filtri

Il segnale elaborato dall'inviluppo è un suono completo e differenziato, con un contenuto timbrico preciso e una propria evoluzione temporale.
La sezione di filtraggio del suono ha il preciso compito di modificare a piacimento il contenuto armonico dell'onda.
I filtri disponibili sono tre: HI-PASS, LOW-PASS e BANDPASS; i tre filtri sono selezionabili contemporaneamente e la risultanza di HP e LP produce un quarto filtro chiamato NOTCH.
Prima di entrare nell'esame approfondito dei filtri è necessario parlare del CUTOFF, che è la misura della frequenza di lavoro del filtro: questo valore può variare da 30 hertz a 12 kilohertz.
Hi-pass filter lascia inalterate le frequenze sopra il cutoff, mentre attenua di circa 12 db per ottava tutte le frequenze sotto il cutoff.
Low-pass filter lascia invece inalterate le frequenze sotto il valore di cutoff, mentre attenua di 12 db le altre sopra.
Possiamo capire meglio questi due filtri, immaginando una

ipotetica finestra delle frequenze, con le basse a sinistra e le alte a destra; il filtro Hi-Pass è una tenda che si chiude da sinistra verso destra, mentre il filtro Low-Pass si chiude da destra verso sinistra. Il valore di cutoff è quindi la misura dell'apertura della nostra ipotetica tenda, che, bisogna ricordare, non chiude, ma filtra le frequenze.
Il **filtro bandpass** lascia invece passare un anello di frequenze attorno alla frequenza del cutoff, attenuando tutte le altre.
L'ultimo filtro è il **notch**, la risultante di HP e LP, che filtra solo un anello di frequenze attorno al valore del cutoff, lasciando inalterate tutte le altre.

Resonance

La risonanza è un effetto che mette in rilievo le frequenze strettamente vicine a quella del CUTOFF. La sua regolazione è parametrica e il suo range è (0-15).
Tutto quello che è stato descritto finora riguarda singolarmente tutte e tre le voci, però il SID è provvisto di altri comandi. Questi interessano particolarmente le relazioni possibili tra le tre voci e le loro combinazioni e influenze.
Infatti è possibile ottenere nuove elaborazioni del suono di un oscillatore, con l'intervento esterno di un altro canale. Gli effetti ottenibili diventano poi rilevanti con l'uso multiplo delle tre voci, in polifonia o in monofonia.

Sync

Rilevante è la funzione di SYNC, che permette la sincronizzazione della frequenza fondamentale dell'oscillatore 1 con quella del 3.
Questo permette di variare la frequenza dell'oscillatore 1 alla frequenza del 3 e di produrre delle complesse strutture armoniche che danno corposità al suono risultante.

Ring

Anche la RING modulation lavora con gli oscillatori 1 e 3, ma in questo caso agisce rimpiazzando l'onda triangolare con una combinazione dell'oscillatore 1 e del 3. L'onda sonora risultante è ricca di strutture non-armoniche che invece di saturare il suono, come con il SYNC, lo rendono più brillante e metallico, così che la RING MOD. è molto usata per riprodurre suoni di campane o altre percussioni metalliche.
È possibile fare ancora moltissime cose con l'aiuto dell'oscillatore n. 3: è possibile associare dei valori numerici alle 4 forme d'onda fondamentali, ed è possibile leggere questi valori usandoli nella programmazione sonora come dati in tempo reale.
Nel SID è possibile applicare questi valori a diversi parametri delle due voci restanti; avremo quindi una modulazione programmata da un L.F.O. (Low Frequency Oscillator) impersonato dal canale n. 3. I parametri che si possono modificare sono: il PITCH, il CUTOFF, il PW (pulse width), che è la misura della rampa dell'onda pulse.
L'effetto del L.F.O. è programmabile: si modifica la velocità, l'intensità di azione sul generatore di suono e la forma d'onda prescelta.
L'azione sul PITCH provoca una modulazione di frequenza con cui si può ottenere il "vibrato".
Il CUTOFF viene modulato e varia l'azione dei filtri in tempo reale, modificando il tono e il colore del suono. La profondità della rampa dell'onda pulse PW viene anch'essa modulata in tempo reale, variando l'effetto "nasale", tipico dell'onda PULSE.
Un'altra importante influenza che l'oscillatore 3 ha sui tre parametri prima accennati deriva dalla possibilità di modulare i valori con l'uso del generatore di inviluppo.
In pratica si può modificare con un A.D.S.R.: il PITCH, il CUTOFF e il PW; e questa è una modulazione utilissima, perché è completamente programmabile, sia come intensità che come durata. Si programma così l'evoluzione dell'effetto.
Il PITCH può essere programmato per ottenere particolari effetti e così anche il PW. Ma la conseguenza più interessante è quella sul CUTOFF, con la completa programmabilità dell'apertura dei filtri.
Con quest'ultima parte abbiamo parlato di tutto quello che è dentro il SID, ed è molto; con la possibilità di accedere a tutti questi controlli e con un po' di pratica si può veramente sfruttare il C64 come un vero synth professionale.
I suoni che si possono creare, con la completa accessibilità del sistema, sono di tale qualità che fa un po' arrabbiare la non ottima qualità del sistema di uscite audio che lasciano andare qualche fruscio di troppo. Però, se non si deve fare un uso professionale dei suoni creati, il sistema svolge perfettamente le sue funzioni. Il problema più difficile da risolvere è appunto quello di riuscire ad accedere a tutte le "facoltà" del SID; perché fino a quando si parla di A.D.S.R., di filtri o di forme d'onda, con poche righe di programmazione BASIC si può toccare con mano.
Ma quando si vuole entrare nella programmazione avanzata, con l'uso delle modulazioni, del L.F.O. o anche della polifonia, bisogna affidarsi al software di mercato o rimboccarsi le maniche per costruire programmi in linguaggio macchina.
In effetti il BASIC è troppo lento e carente di comandi specifici.
La polifonia a sole tre voci non è certamente l'ideale per chi volesse suonare il C64 come un sintetizzatore; ma è abbastanza per chi volesse imparare i primi passi della musica elettronica.
È invece un vero spasso la possibilità di programmare i tre oscillatori separatamente, per costruire sintesi monofoniche. Il risultato, soprattutto sfruttando le modulazioni del terzo oscillatore, è veramente notevole; pensando poi alle dimensioni del SID, c'è da pensare in un ottimo futuro per i moduli di sintesi programmabili via MIDI.
A proposito di interfacce, il nostro SID non ha problemi, oltre alla possibilità di lavorare con synth esterni, grazie alle interfacce MIDI da applicare al C64. Il 6581 è anche provvisto di un sistema per sincronizzare l'oscillatore 1 con eventi esterni.
Però l'Interfacciamento più interessante è quello con l'Uscita EXT IN, che permette di affiancare altri chip 6581 esterni, aumentando la polifonia. Con la stessa uscita è possibile mixare segnali audio esterni con quelli del SID e, meraviglia delle meraviglie, MODIFICARE segnali esterni, come voce, chitarra o piano, attraverso il filtro.
Richiudiamo il magico 6581 e proviamo ad immaginare gli sviluppi presenti e futuri del software e dell'hardware dedicato.
Con l'aiuto di altri fratellini e di una tastiera esterna il SID è ora un piccolo synth di tutto rispetto, ma il suo successo maggiore lo otterrà probabilmente, nel mondo musicale,

come processore di suono a basso costo (anche con hardware esterno).
Sono già in commercio a buon mercato numerosi aggeggini musicali che si basano sul C64 e sulle sue facoltà, come echi digitali, sequencer e altri.
Restando con i piedi per terra, possiamo parlare dell'uso che è stato fatto nei vari software del SID. Ultimamente la parte musicale dei videogiochi non è più fatta dal programmatore del gioco, ma sempre più spesso da musicisti che vengono reclutati dalle case di software.
Il risultato non si è fatto attendere, infatti ora i programmi sono ben forniti di colonne sonore originali, che non si rifanno ai soliti motivi classici (che sono bruttini in trifonia), ma che sono buone sintesi sonore, con buoni motivi musicali: vedi "aerobics dance" "ghostbusters" e molti altri.
Il discorso si potrebbe approfondire parlando dei sintetizzatori vocali e soprattutto dei numerosissimi programmi musicali, ma di questo riparleremo prestissimo.

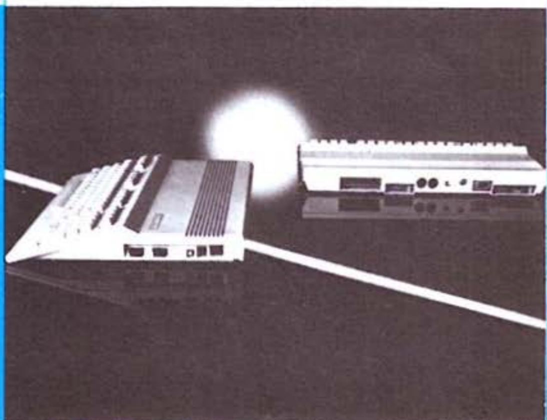
Football Manager come i Pink Floid

Football Manager, il programma della Addictive che vi dà la possibilità di guidare una squadra di calcio alla conquista del duro campionato inglese, non ha solo venduto più di 100.000 copie, ma è rimasto costantemente nelle classifiche di vendita per ben 3 anni: un fatto anomalo, visto che solitamente i programmi rimangono in classifica per poche settimane.
Un record che è paragonabile a quello dei Pink Floid i quali, con il loro disco "The Dark Side Of The Moon", occuparono un posto nelle classifiche per parecchi anni.
Nel frattempo sono stati realizzati altri programmi sul calcio. Gli ultimi due titoli sono FIVE-A-SIDE, dove la partita prevede anche i falli e i calci di rigore, e Bobby Charlton Soccer, dove sono abbinati i tre ruoli di una squadra di calcio: manager, allenatore e calciatore.



Il nuovo C128, fratello maggiore del C64

Presentato al pubblico italiano durante l'ultima edizione dello SMAU, il C128 ha subito appassionato i vecchi e i futuri possessori di un computer della casa rosso e blu.
Dalla linea moderna e slanciata, il C128 vuole essere il tentativo di legare un passato glorioso a un futuro sempre più esigente.
La presenza di tre microprocessori differenti permette all'utente di utilizzare il computer in tre diversi modi.
Il successo mondiale del C64 ha costretto la casa americana a non sottovalutare l'innumerevole quantità di giochi e programmi gestionali prodotti dalle varie software house.
Così, grazie al microprocessore di base, il MOS 8502, il C128 diventa perfettamente compatibile con tutti i programmi del CBM.
La vera novità di questo computer è comunque il sistema di base, che si basa sempre sull'8502.
La memoria di base è di ben 128 Kbyte di RAM, espandibile fino a 512 Kbyte, e incorporato c'è il BASIC 7.0 con oltre 140 comandi e il video a 80 colonne per 25 righe.
La terza faccia del C128 è data dallo Z80 A, il secondo microprocessore contenuto nella macchina. Grazie a questo diffuso microprocessore a 8 bit, si ha la possibilità di utilizzare il CP/M 3.0, con il conseguente accesso all'enorme quantità di programmi che utilizzano questo sistema operativo.
Tutte le periferiche già in commercio per il C64 sono perfettamente compatibili con il C128 e altre dalla linea più moderna sono state aggiunte.
Sono così disponibili il nuovo Disk Drive 1571, con capacità di memorizzare 340 Kbyte, il Monitor 1901 a 13" ad alta risoluzione grafica, che visualizza 40 o 80 colonne, e il mouse, il nuovo sistema per trasmettere le informazioni al computer senza utilizzare la tastiera.
A questo proposito è disponibile il Jane, un pacchetto di



programmi integrati, contenente word processor, foglio elettronico e data base, utilizzabili con l'uso di icone e del mouse.
Nuovo software su misura per il C128 è anche il MICRO ILLUSTRATOR, un programma che sfrutta al massimo le notevoli capacità grafiche della macchina, e SKY TRAVEL, un programma di astronomia, con il quale si possono studiare pianeti, stelle e fenomeni celesti.
Il C128 non è certo un computer rivoluzionario. La Commodore ha voluto realizzare una macchina che rinnovasse il glorioso C64 senza metterlo definitivamente in pensione ed adottando la non nuova soluzione della pluri compatibilità.

Herbert Dummy Run

Sempre per la serie delle avventure della famiglia di Wally Week, questo mese in edicola trovate, per la collana di giochi pubblicati da Jackson Soft Oro, un'avventura di HERBERT, il simpatico figlio un po' dispettoso di WALLY.
I fedeli seguaci di Jackson Soft Oro hanno già conosciuto il simpatico fanciullo in Everyone's a Wally, dove si aggirava a quattro zampe tra le vie della città, combinando brutti scherzi a tutti i personaggi.
In Herbert Dummy Run il giovane protagonista ha finalmente imparato a camminare e, non contento di combinare guai, si perde all'interno di un grande magazzino della città.
WILMA e WALLY, i suoi genitori, sono costretti ad attenderlo nell'ufficio delle persone smarrite, mentre il loro affezionato figliolo è alla ricerca di una strada tra giocattoli, oggetti ed ascensori.
Il vostro compito è proprio quello di aiutare il simpatico pargolo a districarsi tra giochi, videogiochi ed insidie, per raggiungere i genitori nella stanza delle persone smarrite prima della chiusura dei grandi magazzini.



La grafica è sempre stupenda, nello stile di Pigyamarama, e anche le caratteristiche sono simili, con schermi da perlustrare, oggetti da raccogliere e di cui si deve scoprire l'uso, ostacoli da evitare senza perdere troppa energia. La caratteristica originale di Herbert Dummy Run è la presenza di numerosi videogiochi, fondamentali per arrivare alla soluzione finale.
BREAK-OUT, TURTLES o SPACE INVADERS sono alcuni dei videogiochi contenuti, che bisogna affrontare con il giusto spirito e l'oggetto appropriato.
L'unità di misura con cui vi viene indicato il progredire dell'avventura è composta da squisite caramelle.
Non dimenticatevi che per trovare i genitori di Herbert avete tempo dalle 13 fino alle 17,30, ora di chiusura dei grandi magazzini.

Match Point

Prodotto da: PSION
Supporto: cassetta
Prezzo: L. 25.000

Tra i giochi di simulazione sportiva, il tennis e tutti i giochi di questo tipo sono stati tra i primi ad essere tradotti sul computer. Indubbiamente, siamo ormai lontanissimi dai primi videogame, quasi senza alcuna cura grafica e con uno schema di gioco piuttosto povero e scontato. Sebbene il meccanismo e lo scopo del gioco del tennis siano molto semplici, la cura dei particolari grafici e la possibilità di effettuare diversi colpi offrono quel realismo che contribuisce in modo determinante al coinvolgimento dei giocatori. Insomma, un conto è colpire una macchiolina bianca con una paletta stilizzata, un altro è vedere il proprio giocatore fare il servizio, andare a rete, correre indietro a recuperare su un lob dell'avversario.
Con tutto ciò non si può certo dire che il realismo di questo gioco sia assolutamente perfetto. Per esempio, è molto più difficile che nel tennis reale riuscire a dirigere il colpo dove si vuole, sia per quanto riguarda l'angolazione che la profondità. Il servizio poi è quasi completamente guidato dal computer e quindi non è possibile tentare un ace o rischiare una seconda palla. Spesso infine, soprattutto giocando al primo livello di difficoltà, si ha l'impressione che la pallina non venga colpita dalla racchetta, ma dal corpo del giocatore e quindi si è tentati ad andare a ridosso della palla, anche se tutto questo, da un punto di vista tecnico, è assolutamente sbagliato.

Il gioco

Prima di iniziare è possibile scegliere a quale livello di difficoltà giocare e se disputare la partita contro il computer o contro un altro giocatore.
I tre livelli di difficoltà corrispondono ai quarti di finale, alla semifinale e alla finale di un torneo di tennis ipotetico. Come al solito, con l'aumentare del livello di difficoltà aumentano la velocità della pallina e dei giocatori. La prima impressione è che ci si possa accontentare del primo livello per le prime due partite, mentre il terzo richiede dei riflessi da superman.
Le partite vengono disputate tutte al meglio dei 5 set, con tie-break sul 6-6. Il computer è battibile abbastanza facilmente al primo livello, mentre nell'ultimo le cose si complicano notevolmente.

Molto più divertente è confrontarsi con un altro giocatore, perché naturalmente entrano in gioco altri stimoli.
Le regole del tennis sono rispettate in ogni dettaglio: si cambia campo dopo il primo gioco e successivamente dopo ogni due, mentre al tie-break il servizio cambia dopo il primo punto, ogni due volte e il cambio di campo avviene dopo sei punti. Purtroppo, almeno per i primi tempi si potrà constatare un certo vantaggio nel giocare nel campo più vicino al fondo dello schermo: infatti vi si gode una prospettiva che consente una maggiore identificazione con il proprio giocatore.
Rispetto al tennis reale sono piuttosto rari gli errori, cioè scarseggiano le palle chiamate out e quelle che finiscono in rete. Molto gradevole la trovata dei raccattapalle che raccolgono le palline rimaste sul campo sui colpi finiti in rete.
Sul fondo dello schermo è ben visibile il tabellone con il punteggio completo: inoltre il computer provvede a segnalare le fasi decisive con i messaggi di set e match point. Gli altri messaggi del computer si riferiscono alle palle out e alla seconda palla di servizio.

I comandi

I comandi sono molto semplici: durante il gioco è sufficiente avvicinare il giocatore alla pallina in arrivo, quindi premere il bottone del fire; in questo modo il giocatore effettuerà un mezzo giro su se stesso, dando l'impressione di colpire la palla. E' opportuno premere il fire anche per posizionare giocatore e racchetta nel modo più appropriato per ribattere la palla.
Per quanto riguarda il servizio è ancor più facile, perché è il computer stesso che impedisce l'errore: il giocatore che deve effettuare il servizio viene disposto sulla riga di fondo, nella esatta posizione per effettuare la battuta verso destra o sinistra; quando viene premuto il bottone del fire il giocatore lancia la palla in alto e con la seconda pressione la colpisce. In ogni caso ci si accorge immediatamente che anche premendo una sola volta, per alzare la palla, il computer stesso effettua, per così dire, un servizio di default.



Conclusioni

Il gioco è sicuramente gradevole e con un po' di pratica è possibile trovare le tecniche giuste per effettuare colpi spettacolari. Inoltre, grazie anche alla cura grafica dei dettagli, è possibile un forte coinvolgimento, specialmente confrontandosi con un altro giocatore.

Master Op Lamp

Prodotto da: Activision
Supporto: cassetta
Prezzo: L. 30.000

In questo arcade per C64 le qualità determinanti non sono i riflessi, come nei giochi spaziali di combattimento, o la tecnica, come in giochi del tipo di Match Point, Soccer, One on one, ecc., piuttosto è richiesta una notevole capacità di percepire e riprodurre sequenze di note e colori. Non è possibile parlare di tattiche, perché il gioco è talmente semplice che è inutile affannarsi a scoprire trucchi per battere il computer. In effetti, almeno per i primi tentativi, il giocatore avrà bisogno di una buona dose di pazienza e dovrà resistere alla tentazione di lasciar perdere tutto e passare ad altro.
Il gioco per i principianti risulterà infatti piuttosto monotono, in quanto sarà dura riuscire a superare persino la prima prova.

Tre prove

Dato il RUN apparirà per prima cosa l'elenco dei giochi in cui è possibile cimentarsi: SETTE PROVE, RICERCA DEL TRONO e TAPPETO MAGICO.
Il primo è il più facile dei tre e consiste nel percorrere una serie di tunnel a bordo di un tappeto magico, senza cadere e senza andare a sbattere contro le pareti dei tunnel. Questi sono rappresentati da una serie di rombi concentrici, che avanzano verso il tappeto magico, fluttuando verticalmente e orizzontalmente. Una volta usciti dal tunnel si giunge nell'antro del genio, dove bisogna tentare di ricostruire le sequenze che vi suggerisce.
Per ricostruire la sequenza occorre muoversi orizzontalmente lungo la fila di gong e suonare quello corrispondente alla nota visualizzata in alto sullo schermo, prima che questa ricada verso il basso. In questa fase, la più facile delle tre, saranno d'aiuto sia le note che i colori. Per completare questo livello e diventare il padrone del gioco sarà necessario ricostruire una lampada costituita da sette pezzi. Nel secondo gioco, LA RICERCA DEL TRONO, le lampade da ricostruire sono tre, sempre di sette pezzi ciascuna,



ma in questo caso suoni e colori appariranno soltanto alternativamente.
Nei sette antri del primo genio vengono dati colori e suoni, ma i colori spariscono molto rapidamente. Nei sette del secondo genio vengono forniti soltanto i colori, mentre in quelli del terzo sarà possibile ricostruire la sequenza soltanto in base alle note.
Il terzo gioco, il TAPPETO MAGICO, serve esclusivamente per far pratica di volo con il tappeto: è possibile infatti selezionare uno dei 21 tunnel, dal più facile al più difficile, ed esercitarsi continuamente.

I comandi

Durante la fase di volo sul tappeto i comandi sono quelli tradizionali e il joystick si comporta, come al solito, in modo simile alla cloche di un aereo. Nell'antro del genio, invece, serve per spostarsi lungo la serie di gong: inclinando verso sinistra ci si sposta gradualmente verso sinistra, inclinando verso destra ci si sposta gradualmente verso destra, premendo fire ci si sposta di scatto sul gong successivo, nella direzione verso la quale si è orientati, infine spostando il joystick in avanti si suona il gong sul quale si è posizionati.
In qualunque momento del gioco è possibile ritornare al menu delle prove, premendo contemporaneamente RUN/STOP e RESTORE.

Conclusioni

Il pericolo in un gioco di questo tipo è di non riuscire ad avere sufficiente pazienza per passare oltre la prima lampada. Inoltre la carenza quasi assoluta di qualsiasi spunto tecnico/tattico lo rende ancora meno interessante. Complessivamente non si può certo dire che sia un gioco avvincente, perché non offre gli spunti necessari per realizzare delle competizioni.

Psi Warrior

Prodotto da: Beyond
Supporto: cassetta
Prezzo: L. 30.000

Il protagonista, PSI WARRIOR, si trova all'interno di un grosso silos, diviso in numerosi livelli circolari e collegati fra loro da elevatori e da guide inclinate. WARRIOR deve ripulire il silos dalle entità aliene che lo infestano ed infine raggiungere la sorgente di questi mostri e distruggerla. Warrior può muoversi, per mezzo di un piano volante, lungo delle guide magnetiche che attraversano i corridoi dei vari livelli.
Per intraprendere la difficile missione Warrior ha a sua disposizione sia energia ID che energia PSI. La prima fornisce il carburante per tutta la missione, la seconda viene utilizzata per controllare l'emissione di energia ID e per accedere ad alcuni poteri nascosti della mente, come ad esempio la levitazione, il teletrasporto, la visione remota e l'invisibilità.
La levitazione permette di superare alcuni ostacoli altrimenti insormontabili o di togliersi dai pasticci quando si viene attaccati da un numero troppo elevato di entità. Il teletrasporto consente di ritornare rapidamente in un centro di propulsione, per fare rifornimento. La visione remota è molto utile per esplorare i vari livelli senza rischiare di incorrere in trappole e senza doversi curare degli ostacoli. L'invisibilità confonde le entità PSI e può essere determinante per sfuggire a qualche agguato.

I nemici

Le entità che occupano il silos e che si possono incontrare durante la missione sono di tre diversi tipi:
1) PSI: questi sono gli abitanti più intelligenti e pericolosi del silos; appaiono sullo schermo come ammassi di luci lampeggianti con sagoma allungata verticalmente.
2) ID: sono meno intelligenti degli PSI, ma possono essere ugualmente pericolosi, se il loro grado di energia è elevato.
3) PUPAE: queste sono entità umane liberate dal corpo e vengono ancora attratte dagli uomini. Possono essere pericolose, se ci si inciampa, inoltre, se per errore il Warrior dovesse tentare di catturarne una, tutte le altre gli si rivolteranno contro.
Quando si incontrano creature PSI o ID è necessario catturarle, lanciandogli addosso le reti e assorbendone l'energia associata. Bisogna assolutamente evitare di essere colpiti da queste creature, altrimenti saranno loro a ridurre l'energia vitale dello PSI-WARRIOR.
La sorgente delle entità PSI e ID si trova nel livello più profondo del silos, quindi è là che PSI-WARRIOR deve cercare di dirigersi il più rapidamente possibile. Ma attenzione, è indispensabile raggiungere la sorgente con una buona riserva di energia, perché questa esercita una forte influenza negativa sulla mente di tutti gli esseri umani che si avvicinano ad essa, e ne è quindi necessaria una grande quantità per mantenere il controllo del Warrior. Inoltre, a differenza delle entità, per distruggere la sorgente bisogna lanciarle contro tre reti e quindi è fondamentale poter manovrare con facilità il Warrior.

COMMODORE TI FA U

Prendi monitor o disk drive...



Grande Operazione "Fai di piú col tuo Commodore."

Monitor Commodore e disk drive 1541 sono le periferiche che ti permettono di utilizzare al meglio il tuo computer Commodore e di prepararti per il futuro.

Con il disk drive Commodore, memorizzi piú velocemente i dati su dischetti standard e puoi caricare i programmi in un attimo.

Con il monitor a colori ad alta definizione d'immagine puoi evidenziare le tue qualità di programmatore, utilizzando al massimo le capacità sonore e grafiche del tuo computer.

Monitor e disk drive oggi convengono, perchè hanno un prezzo eccezionale, come d'abitudine Commodore. Inoltre, proprio per dimostrarti quanto puoi fare di piú con queste periferiche, Commodore ti fa un regalo speciale.

Acquistando monitor o disk drive entro il

N REGALO SPECIALE



30 novembre 1985, hai insieme, già compresi nel prezzo come regalo, il favoloso computer-rack creato per ospitare il tuo sistema Commodore, oppure i formidabili sci del team Commodore Adventure.

Due opportunità incredibili ma vere. Vai subito al tuo negozio Commodore e approfitta dell'offerta "Fai di più col tuo Commodore".

sci: cm 175 / 195 - rack: cm 90×90×70

Commodore

I comandi

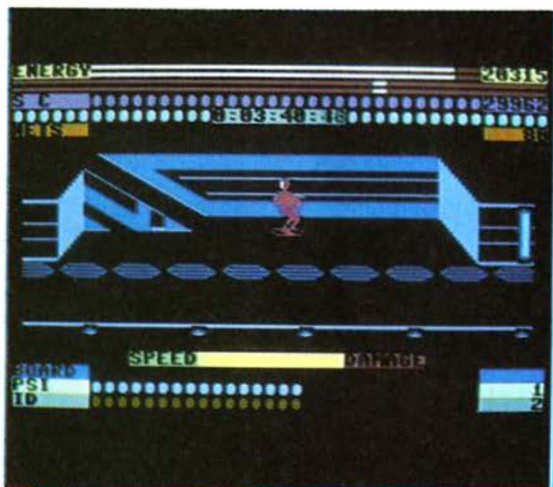
E' possibile giocare anche senza joystick ed utilizzare la tastiera anche per il movimento del Warrior e per il fuoco; quindi joystick in porta 2 oppure:

CTRL	= alto
SHIFT sinistro	= basso
CLR/HOME	= sinistra
INST/DEL	= destra
SHIFT destro	= fuoco/salto
RUN/STOP	= pausa
@@@	= inizia una nuova missione
Tasto Commodore	= registra le coordinate
F1	= teletrasporto
F5	= levitazione
F3	= visione remota
F7	= invisibilità

Il pulsante di fuoco interrompe l'invisibilità e la levitazione.

Conclusioni

Il gioco è abbastanza ricco di trovate, alcune delle quali originali, e la cura grafica dei particolari non è male. Scarseggiano un po' gli effetti sonori, mentre abbastanza criticabile ci sembra la definizione grafica delle entità come ammassi di luci lampeggianti; indubbiamente si sarebbe potuto trovare di meglio.
Azzeccato invece il comando che consente di salvare la posizione del Warrior su disco o su nastro. Infatti le missioni possono durare anche molto tempo, se si è particolarmente bravi o se non si conosce sufficientemente il percorso ottimale per scendere nel silos, quindi poter sospendere la partita senza dover ricominciare daccapo è quasi indispensabile.
Per quanto riguarda la velocità, questa normalmente non è elevata e dipende molto dalla capacità del giocatore di mantenere intatte le proprie riserve di energie e di non rovinare la piattaforma di volo con rovinose cadute.
Tutto sommato, quindi, PSI WARRIOR richiede sia prontezza di riflessi nel reagire di fronte agli attacchi delle entità, ma soprattutto è fondamentale elaborare una strategia che sfrutti tutti gli strumenti di cui si dispone, alcuni dei quali nelle prime partite sembrano passare in secondo piano. La visione remota ad esempio, che consente di guardare in tutto il silos, permetterà a chi non conosce ancora tutti i livelli di trovare la via più breve per giungere fino alla sorgente delle entità, che è in definitiva lo scopo ultimo del gioco.



On Field Football

Prodotto da: Activision
Supporto: cassetta
Prezzo: L. 25.000

Tra tutti i giochi sportivi per home computer certamente questo è uno tra i più ambiziosi.
ON FIELD infatti, almeno secondo il manuale, si rivolge a un pubblico di ROOKIE, cioè di principianti, tuttavia già dalle prime battute appare chiaro che senza conoscere almeno le regole più importanti del football americano non si riuscirà neppure ad iniziare a giocare. Inoltre, se si vuole combinare qualcosa di buono contro il computer è necessario per lo meno avere in testa alcuni schemi di gioco fondamentali. Naturalmente, questo videogame non è di quelli che: "pronti, via!", e si può sperare di fare il record. Per prima cosa, al contrario di tanti giochi analoghi, il computer è molto più bravo e molto più esperto di qualsiasi Rookie, quindi non abbattetevi, se le prime volte subirete delle batoste paurose. Dobbiamo pensare che al contrario degli altri giochi sportivi, per noi europei, in generale, anche l'originale presenta delle difficoltà. Tutti infatti sappiamo bene cos'è una volet, una triangolazione o un terzo tempo, ma quanti di noi hanno mai sentito parlare di extra-points, safety, o kickoff?
Il primo passo, dunque, è riuscire a capire bene le regole del gioco reale, perché anche se questo è solo un videogame, le difficoltà più grosse derivano proprio dal suo realismo. Il compromesso necessario tra realismo e giocabilità era particolarmente difficile, dato che il football americano, per sua propria natura, necessita sia di rapidità di esecuzione che di strategia. Il fatto stesso di dover obbligatoriamente gestire i passaggi e le ricezioni implica la possibilità di poter manovrare più di un giocatore alla volta. In effetti questa potrebbe sembrare la difficoltà più grossa in quanto, mentre il QB sta effettuando il lancio il WR deve poter raggiungere la palla. Naturalmente, le prime volte sarà già un successo riuscire a lanciare prima di vedere il QB schiacciato dai linebacker avversari.

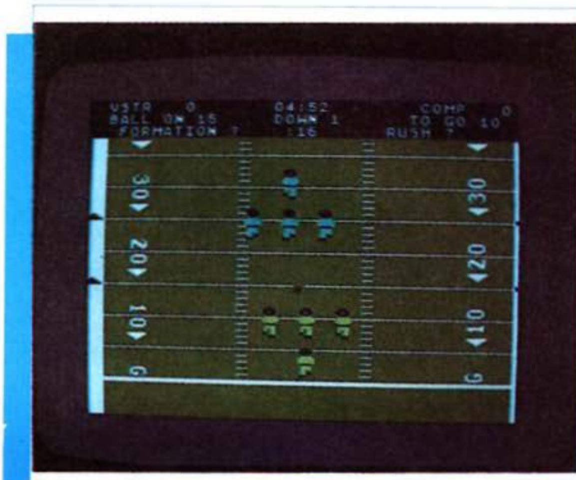
Ma prima di vedere come vincere occorre dare un'occhiata alle regole del gioco: per fare questo diamo per scontato che il lettore conosca un po' l'originale.

La formazione

Per prima cosa, ogni squadra è formata solo da 5 giocatori. E' possibile giocare contro il computer o contro un altro giocatore. Prima di iniziare la partita ogni giocatore può scegliere la propria formazione di attacco, in base alle caratteristiche di alcuni giocatori chiave.
E' possibile scegliere tra un QB che possiede una grande precisione nei passaggi, ma che però non è molto mobile, e tra un altro che al contrario non è un eccellente passatore, ma che però, se riesce a trovare un varco nella linea difensiva, è capace di incunearsi grazie alla sua velocità. Anche per gli altri due ruoli dell'attacco, il WD (wide receive) e TE (tight end), è possibile scegliere quello che possiede le caratteristiche più adatte agli schemi che si intendono realizzare.
Proprio questa è la chiave del gioco: nessun giocatore è, in assoluto, più forte di un altro, ma è più o meno adatto a realizzare certi schemi. Per esempio, è sbagliato tentare di realizzare dei passaggi di 40 yards, avendo scelto un QB poco preciso e un WR velocissimo, ma che non dispone di una grossa presa.
Tutto questo comunque risulterà chiaro man mano che si proverà a giocare, e a perdere, contro il computer, oppure guardando con attenzione il demo del gioco.
Il tempo di una partita può essere molto lungo: infatti è possibile scegliere se disputare quarti di gioco da 15 minuti o da 5. Naturalmente, sarà opportuno iniziare con tempi brevi, perché non bisogna dimenticare che il football viene disputato con tempi semi effettivi, vale a dire che il cronometro si ferma quando la palla va fuori.

Il gioco

Il gioco inizia come al solito con il kickoff. Quando il giocatore che ha ritornato il calcio è stato atterrato inizia la prima azione offensiva dal punto in cui è stato fermato. Si hanno 4 tentativi per avanzare di 10 yard e in ogni momento è possibile optare per un calcio tra i pali. Prima di iniziare l'azione offensiva è necessario scegliere lo schema di attacco e la direzione del passaggio. Nella parte alta dello schermo comparirà, tra le altre cose, l'opzione selezionata. E' possibile scegliere tra: TE, WR, QB, HB, C. Ognuna di queste scelte comporta un diverso movimento dei 5 uomini sul terreno e quindi una volta fissato lo schema deve essere eseguito, compatibilmente con le scelte difensive fatte dall'avversario. E' importante sottolineare che la squadra in attacco si trova sempre nella parte più bassa dello schermo, cioè l'attacco è sempre rivolto verso l'alto dello schermo. Con questo accorgimento, infatti, è possibile evitare gli errori indotti da un cambio di prospettiva durante il gioco.
Anche chi si difende può scegliere il proprio schema: è possibile decidere sulla posizione del CB secondario o se effettuare un blitz sul QB avversario.
Ogni squadra ha a disposizione tre time-out ogni due quarti di gioco e 30 secondi di tempo per scegliere la propria strategia offensiva, scaduti i quali viene penalizzata di 5 yard.



Conclusioni

Il gioco è veramente molto realistico, tanto da risultare anche un po' complicato. Tuttavia, appena si fa pratica è veramente emozionante riuscire ad elaborare e a realizzare gli stessi schemi che si vedono in televisione nelle partite della NFL. Per i principianti è consigliabile osservare attentamente il demo e provare a realizzare tutti gli schemi prima di mettersi a competere veramente. E' inoltre utile impratichirsi nel lancio del QB che non è molto semplice. Superate le difficoltà iniziali, questo sarà senza dubbio un gioco che farà fare le ore piccole a non pochi appassionati di football americano.

Pit Stop II

Prodotto da: Epyx
Supporto: disco e cassetta

Nel ricco panorama dei giochi di simulazione sportiva realizzati per il C64, Pit Stop II è da considerarsi sicuramente tra i meglio riusciti in assoluto.
Realizzato con una buona grafica e una fedele riproduzione di una pista automobilistica, questo gioco riesce sicuramente nel difficile compito di interessare ed appassionare anche i più pignoli ed esperti tifosi di Formula 1.
Se fino ad ora potevate gareggiare con un vostro amico in un video-gran premio automobilistico solo comparando i tempi realizzati in due gare effettuate separatamente,

con Pit Stop II vi potrete finalmente misurare sullo stesso circuito, e contemporaneamente, con chiunque vi sfidi.

Due schermi

Durante la gara lo schermo del computer è diviso in due parti, una superiore e una inferiore, ciascuna delle quali descrive il punto di vista ottico di due distinte macchine da corsa, ciascuna guidata da un diverso pilota.
Ovviamente, se le due macchine si trovassero abbastanza vicine durante la gara, oltre all'eventuale quanto pericoloso contatto fisico, sarà possibile avere anche il divertente contatto visivo, nel senso che il pilota più attardato dei due potrà vedere sulla sua parte di schermo il retro della macchina avversaria.
Una volta caricato il gioco in memoria vi sarà chiesto il numero dei giocatori, la pista prescelta per la gara, il numero dei giri da compiere prima del traguardo e il livello di difficoltà.
Selezionando l'opzione per un solo giocatore, sarà possibile concorrere contro la macchina guidata dal computer, invece che contro un vostro amico.

Sei circuiti

Pit Stop II ha in memoria 6 diversi circuiti mondiali: oltre a poterli percorrere separatamente, scegliendo via via quello più adatto per la gara, potrete chiedere di disputare il campionato del mondo, cioè la partecipazione a tutti e 6 i Gran premi, con la memorizzazione dei punti ottenuti via via da voi, dal vostro eventuale compagno di gioco e dai piloti che, in ogni gara, concorrono insieme a voi guidati dal computer e che, anche se abbastanza lenti, possono rubarvi punti preziosi.
Inoltre, sempre prima di iniziare una gara o un campionato, potrete scegliere se compiere 3, 6 o 9 giri di pista e se usare la guida più facile, quella media oppure quella più impegnativa.

La gara

In gara avrete la possibilità di pilotare la vostra auto attraverso il joystick: muovendolo a sinistra o a destra la vostra auto curverà di conseguenza, mentre spingendo in avanti la leva, anche durante una curva, la velocità aumenterà fino a raggiungere il massimo possibile.

Accelerando, si deve tener presente che se si tiene premuto il pulsante del joystick, la macchina ottiene una migliore ripresa e una maggiore velocità, anche se aumenta il consumo di carburante.
Tirando indietro la leva, invece, si ottiene una decelerazione della velocità fino al completo arresto della vettura.
In gara, la grafica del gioco è tridimensionale, simile a quella di POLE POSITION.
Sulla propria parte di schermo ogni concorrente può osservare la rappresentazione grafica schematizzata del circuito, potendo tenere così sott'occhio la posizione esatta della vettura all'interno dello stesso.
La macchina è disegnata in primo piano, mentre il suo movimento sulla pista viene simulato con una bella dinamica.

Una volta in pista, da tener ben presente sono sia il consumo di carburante che lo stato delle gomme.
Alla partenza, infatti, la vettura comincia con il serbatoio pieno, ma dopo solo 2 o 3 giri la situazione del serbatoio diventa critica.
Le gomme della macchina, se urtate contro il bordo della pista o contro un'altra vettura, si consumano sempre maggiormente, e il loro stato è indicato dal colore che assumono: completamente nere sono in perfetto stato, rosse sono in crisi e una volta bianche rischiano l'esplosione.

I box

Malgrado tutte le attenzioni, mantenere le gomme integre è molto difficile, per il fatto che la macchina, soprattutto nei livelli di difficoltà maggiori, tende a sbandare visibilmente in curva, richiedendo grande esperienza e bravura da parte del pilota per non andare sui cordoli.
Con i livelli di difficoltà più elevati, inoltre, il consumo delle gomme è più veloce.
Se ci si trova in difficoltà con il carburante o con i pneumatici, è possibile la sosta ai box, durante la quale il proprio team, manovrato sempre dal joystick, si occupa della sostituzione delle gomme e del pieno di benzina.
Una gara con più di 3 giri richiede parecchie soste per il rifornimento.
Prima del traguardo ogni gara riserba diversi pericoli: un errore di perizia può portare all'esplosione di una gomma o alla fine della benzina contenuta nel serbatoio, con la conseguente eliminazione dalla gara e con l'evidente danno in fatto di punti, soprattutto se durante un campionato.
Inoltre una sosta prolungata ai box, o una diminuzione della velocità per il pericolo di rimanere a secco o senza una ruota, può farvi perdere preziosi secondi, sufficienti a volte a decidere l'esito di un gran premio, che vi auguro essere, per un maggior divertimento, sempre incerto fino alla fine, ma sempre il più positivo possibile.



Shadowfire

Casa produttrice: Beyond
Distribuito da: Lago
Supporto: cassetta
Prezzo: L. 30.000

Le tribù di pirati che infestavano i mondi alla periferia delle galassie sono state sconfitte irrimediabilmente, ed ora non c'è più nessun ostacolo per il diffondersi della civiltà anche nei pianeti più sperduti. Tutto sembra quindi andare bene, ma non è così.
Un malvagio generale chiamato Zoff ha disertato, con un manipolo di fedeli, dall'esercito dell'Unione ed è riuscito a far perdere le sue tracce, rifugiandosi in qualche angolo dell'universo con un'astronave per perfezionare i suoi piani di conquista dell'intera galassia.
Il suo perfido piano è proseguito senza intoppi: Zoff è riuscito a catturare l'ambasciatore Kryxix, che nasconde, in una microcapsula collocata nella sua colonna vertebrale, i piani di costruzione di una potentissima astronave da guerra, in grado di spostarsi da un pianeta all'altro alla velocità della luce. Il pericolo è grande, poiché Zoff, entrando in possesso di tali piani, potrebbe costruire una flotta da guerra di potenza inaudita e realizzare così il suo tremendo progetto.
L'Unione Planetaria, che ormai da secoli ha affermato il suo potere su tutta la via Lattea, vuole annullare questo tentativo per ristabilire pace e tranquillità.
Alcune spie hanno individuato il quartier generale di Zoff e il grande Consiglio dell'Unione Planetaria ha affidato alla squadra Enigma, un gruppo di cinque agenti segreti pronti a tutto, il compito di liberare l'ambasciatore Kryxix, catturare Zoff e distruggere l'astronave dei ribelli con una bomba ad altissimo potenziale.
A voi dunque il compito di guidare la squadra Enigma attraverso questa missione quasi suicida.

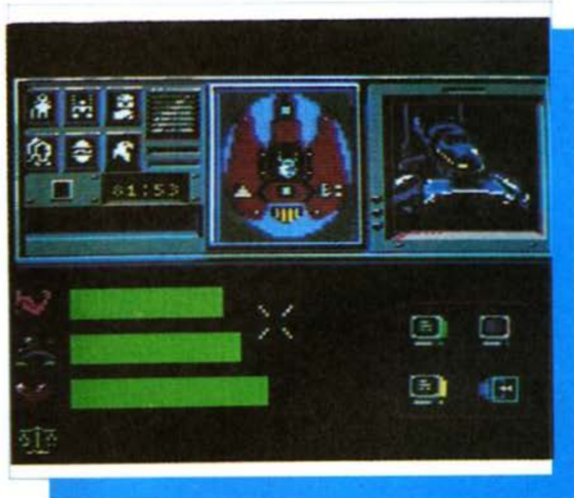
Il gioco

Terminato il caricamento del programma, e dopo aver scelto con i tasti funzione il tipo di comando da utilizzare tra tastiera, joystick digitale, joystick analogico e penna ottica, avete accesso al menu di partenza, con il quale è possibile iniziare una nuova partita, continuarne una già cominciata, caricare o salvare una situazione di gioco.
La grossa novità presente in questo programma, che lo differenzia da tutte le altre produzioni, è che la selezione non avviene attraverso la pressione di un tasto o cose simili, ma attraverso il posizionamento di un cursore che potete muovere con il joystick o con i tasti.
Questo sistema di selezione non si ferma soltanto a questo menu, ma è parte fondamentale dello stesso gioco, che è stato concepito come un adventure senza testo, in cui le azioni avvengono mediante l'opportuna scelta a mezzo di icone, cioè simboli grafici che possono rappresentare ad esempio il combattimento, il movimento, ecc.
Non dovrete fare altro quindi che muovere il cursore sulla icona che volete attivare e premere FIRE per confermare la scelta.

Effettuate le selezioni iniziali potete passare al menu del personaggio, dove potrete scegliere l'agente che intendete utilizzare al momento.
Scelto il personaggio vi verranno visualizzati sotto forma di istogrammi la sua forza fisica, l'agilità, la resistenza psicologica e il peso degli oggetti che il personaggio trasporta, peso che non deve essere superiore alla sua forza fisica.

I terminali

Sulla destra vi sono tre icone a forma di piccoli terminali, di colore rispettivamente giallo, verde, rosso.
Scegliendo l'icona gialla accederete al menu degli oggetti:



in basso a sinistra potrete vedere gli oggetti che si trovano sul posto, mentre al centro sono presenti gli oggetti che il vostro personaggio sta trasportando.
Sulla destra sono visualizzate quattro icone, che servono per "attivare" un oggetto (alcuni oggetti hanno bisogno di questo comando per diventare operativi) per prendere o lasciare un oggetto, mentre l'ultima serve per tenere un'arma pronta per l'uso.
Mediante l'icona verde si accede al menu del movimento, nel quale potrete scegliere in quale direzione muovere il vostro personaggio: il tempo che ciascun personaggio impiega per completare la mossa è proporzionale alla sua agilità.
Il tempo è un fattore molto importante, poiché avete soltanto un'ora e 40 minuti di tempo reale per liberare l'ambasciatore Kryxix ed evitare il fallimento della vostra missione, fallimento che darebbe così la possibilità al generale Zoff di conquistare la galassia.
Infine il terminalino rosso serve per accedere al menu di combattimento, che contiene sulla destra tre icone; una freccia rivolta a destra, una doppia freccia e una freccia rivolta a sinistra.
Durante un combattimento, scegliendo la prima icona si attacca il nemico, scegliendo la seconda combattete dove vi trovate, mentre con la terza vi ritirate. E' indispensabile indicare le direzioni di attacco e di fuga, scegliendole col cursore al centro dello schermo.
Per poter sfruttare al meglio le capacità della squadra Enigma è consigliabile leggersi la lunga descrizione dei personaggi presenti nel manuale delle istruzioni, curato molto bene nei minimi particolari e tradotto anche in italiano dalla Lago, la casa importatrice del gioco.

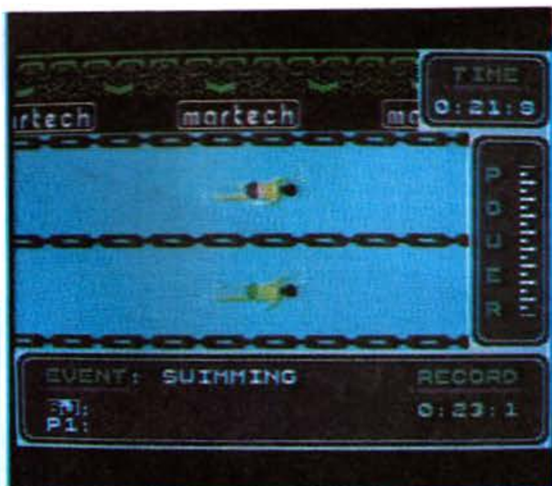


Conclusioni

Shadowfire appartiene a quella categoria di giochi che costituiscono dei veri e propri gioielli di programmazione. L'aver reso un genere di giochi come gli adventure (che erano giocabili solo da chi aveva una buona padronanza della lingua inglese e, perché no, anche una buona immaginazione) alla portata di tutti, con il modernissimo sistema di icone, non può che far onore alla Beyond, gloriosa casa inglese di software. È un acquisto da fare! Oltretutto il gioco viene venduto nella strana confezione contenente sulla cassetta da un lato la versione per lo Spectrum e dall'altro per il C64, due computer tra i più diffusi in Europa.

Brian Jacks Super Star Challenge

Prodotto da: Martech
Supporto: cassetta
Prezzo: L. 25.000



Obiettivo di questo nuovo gioco sportivo è sfidare e provare a battere Brian Jacks, un famoso lottatore di judo, molto conosciuto in Inghilterra grazie ad alcune trasmissioni televisive.
BRIAN JACKS SUPER STAR CHALLENGE mette alla prova la vostra abilità con quella del famoso asso in 8 varie discipline.
Volendo, potete giocare anche con un altro concorrente.

Alcune discipline, come la canoa, il ciclismo, i 100 metri e il nuoto richiedono la solita velocità nello smanettare sulla leva del joystick.
Nel noto il realismo è tale che non dovete dimenticarvi di respirare, per evitare di perdere secondi preziosi.
Una gara abbastanza particolare è il football, una specie di rigore con partenza lanciata, dove prima di calciare in porta dovete dare dimostrazione di abilità, dribblando degli ostacoli.
Le ultime tre discipline sono il tiro con l'arco, le flessioni al suolo e le flessioni alle parallele.
Il punteggio di ogni disciplina viene assegnato in proporzione a quello ipotetico ottenuto dal campione Brian Jacks.
Il gioco non brilla per originalità, anche se i patiti dello sport apprezzeranno sicuramente queste otto nuove prove.
Logica conclusione della competizione è la premiazione finale, sempre gratificante, con tanto di podio per i due concorrenti e naturalmente per Brian Jacks.

Spooks

Prodotto da: Mastertronics
Distribuito da: Mastertronics
Supporto: cassetta
Prezzo: L. 8.400

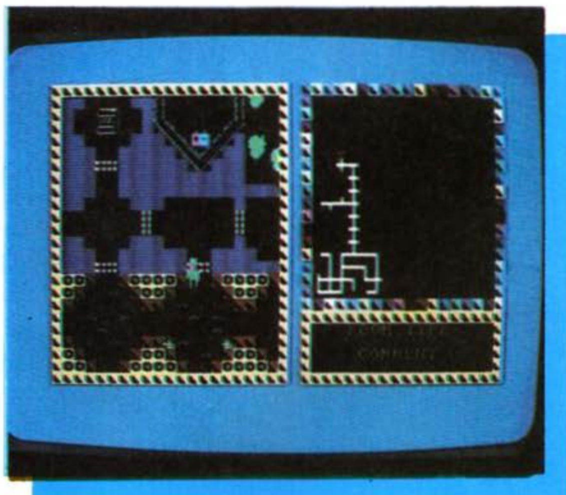
I videogiochi si evolvono e iniziano a nascere i tipi ibridi. E' il caso di Spooks, una miscela tra arcade, platforme game, adventure e gioco a labirinto.
Palcoscenico del gioco è il castello di una prozia, che ha il pallino per il soprannaturale. Vostro compito è quello di recuperare le 8 scatole musicali sparse nell'area del castello e quindi suonare la marcia funebre ed esorcizzare il luogo.
Ogni tocco di uno dei fantasmi del castello vi sarà mortale.
Il gioco inizia nel punto più basso a sinistra nel labirinto.
A fianco dello schermo dove avviene lo "scroll" dell'azione c'è la piantina della casa oppure, se richiamato con il pulsante di fuoco, il menu principale.
In questo caso potete scegliere tra 10 opzioni particolari, come raccogliere l'oggetto trovato, abbandonarlo, terminare il gioco, suonare le scatole musicali trovate, indossare, mangiare, lanciare o accendere qualcuno degli oggetti raccolti ed infine lasciare un segnale di riferimento.
Potete trasportare un massimo di tre oggetti. Ognuno ha una funzione specifica, compresa quella di poter essere usati come peso morto da lanciare contro i fantasmi.

Vi mantiene in vita una quantità di energia che varia da 100 in giù. Il suo valore appare insieme al numero di scatole musicali trovate quando richiamate il menu principale.
"La guida per il profano al mondo degli spiriti" è un libro fondamentale, che vi è di valido aiuto per risolvere il vostro difficile compito.
Tra le pagine si legge tra l'altro: "Non dimenticatevi che i fantasmi compaiono alla mezzanotte".
Questo è un valido consiglio per utilizzare gli orologi sparsi nelle varie stanze che devono essere regolati a dopo la mezzanotte, rendendo così più difficile la comparsa dei fantasmi.
Altri oggetti possono essere utilizzati per superare le acque del lago o per aprire porte insuperabili. Il cibo può essere molto utile per aumentare l'energia, anche se potete trovare quello avvelenato.

Quando trovate un musical box dovete dirigervi verso l'uscita. Automaticamente la scatola verrà depositata e se scegliete l'opzione "PLAY TUNE" nel menu, sentirete il brano della marcia funebre contenuta nella scatola.
Man mano che avanzate scoprite la pianta del castello.
Se raccogliete una mappa, apparità la pianta per esteso e vi sarà molto utile per individuare l'uscita.
Anche il vestiario ha uno scopo preciso e cioè di proteggervi da certi sbalzi di temperatura.

Conclusioni

SPOOKS è un gioco molto avvincente, che richiede, oltre all'intuito tipico degli adventure game, anche una certa abilità, come negli arcade.
Un gioco completo, che vi bloccherà parecchio nel tentativo di sentire quella benedetta marcia funebre.



SUPERCOMMODORE

LIBRI

CEDOLA DI ORDINAZIONE - LIBRI
da compilare e spedire in busta chiusa a
J.soft – Viale Restelli, 5 - 20124 Milano - Tel. 6888228

IL COMMODORE 64 IN FAMIGLIA	cod. AESC022	L. 20.700
ASSEMBLER PER 6502 E 6510	cod. AROK005	L. 13.500

Ordino i seguenti libri per un importo totale di L. ....+ L. 2.000
come contributo fisso per spese di spedizione

Cod. ....Cod. ....Cod. ....

Cod. ....Cod. ....Cod. ....

☐ Contanti allegati ☐ Assegno allegato n° ....

....

☐ Ho spedito l'importo a mezzo vaglia postale

☐ Ho versato l'importo sul cc. n° 19445204 intestato a *J.soft* - Milano

☐ Pagherò in contrassegno al postino al ricevimento dei volumi (valido solo per i soci in Italia)

Nome ....

Cognome ....

Via ....

Città .... C.A.P. .... Prov. ....

Se richiesta fattura:
Cod. F. e P. Iva ....

Data ....

Firma ....

Per i residenti all'estero: pagamento anticipato (vaglia o versamento su ns. ccp)

SUPERCOMMODORE è bello, però... (ovvero suggerimenti, idee, critiche, richieste e tutto ciò che vi passa per la testa).

....

....

....

....

....

....

....

Nome ....

Cognome ....

Via .... C.A.P. ....

Città .... Tel. ....

SUPERCOMMODORE

La rubrica INPUT/OUTPUT è gratuita ed aperta a tutti i lettori. Chi desidera comprare, vendere o cambiare hardware o software può inviare il tagliando a J.soft - Viale Restelli, 5 - 20124 MILANO

INPUT/OUTPUT

☐ COMPRO ☐ VENDO ☐ CAMBIO

☐ VIC 20 ☐ C 16 ☐ C64 ☐ PERIF. ☐ SOFTWARE

....

....

....

....

....

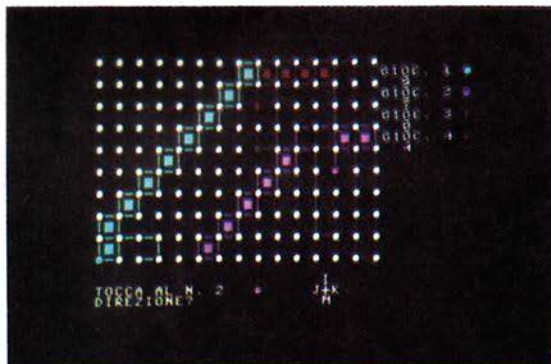
....

Nome ....Cognome ....

Via .... C.A.P. ....

Città ....Tel. ....

```
40 CL(1)=3:CL(2)=4:CL(3)=11:CL(4)=9
                                    :rem  82
50 P(1)=1360:P(2)=1358:P(3)=1440:P(4)=143
   8:CO=54272                       :rem 150
60 FORX=1TO4:P(X)=1024+INT(RND(1)*15)*2+I
   NT(RND(1)*10)*80:NEXT            :rem  61
70 GOTO560                          :rem  59
80 PRINT"{CLR}";:FORX=1TO10:FORY=1TO15:PR
   INT"Q ";:NEXT:PRINT:PRINT:NEXT
                                    :rem 122
90 GOSUB630:PRINTTAB(22);"{GIU'} I{GIU'}
   { 2 SIN}J+K{GIU'}{ 2 SIN}M"      :rem   1
100 QF=1:FORPL=1TO4:P=P(PL):CL=CL(PL):Q=0
    :FL=0                           :rem  79
110 P1=P:GOSUB380:P2=SI:LF=0        :rem 195
120 GOSUB470:GOSUB630:PRINT"{ 2 GIU'}TOCC
    A AL N."PL;:POKE646,CL:PRINT"
    { 2 SPAZI}Q{WHT}"               :rem  29
130 PRINT"{ 21 SPAZI}"              :rem 101
140 GOSUB380                        :rem 176
150 IN=IN(SI,PL):IFIN=0THENGOSUB290
                                    :rem 205
160 GOTO500                         :rem 101
170 IFABS(IN)=1THENPOKEP+IN,67:GOTO190
                                    :rem 105
180 POKEP+IN,66                     :rem 114
190 POKEP+CO,1:POKEP+CO+IN*2,CL:POKEP+IN+
    CO,CL                           :rem 117
200 P(PL)=P+IN*2:GOSUB410           :rem 201
210 IF(S1=15)AND(PEEK(X+D)=32)THENPOKEX+D
    ,160:POKEX+D+CO,CL:B(PL)=B(PL)+1
                                    :rem  94
220 IF(S2=15)AND(PEEK(X-D)=32)THENPOKEX-D
    ,160:POKEX-D+CO,CL:B(PL)=B(PL)+1
                                    :rem 102
230 P=P(PL):GOSUB520                :rem 176
240 GOSUB380:IFP1=PANDP2=SITHENLF=1:Q=9
                                    :rem 230
250 IFQ=>9THEN270                   :rem 243
260 Q=Q+1:GOTO140                   :rem 219
270 IFLF=0THENQF=0                  :rem  90
280 NEXTPL:GOTO640                  :rem 130
290 GOSUB630:PRINT"{ 3 GIU'}DIREZIONE?":P
    OKEP+CO,CL:POKE198,0            :rem 114
300 IFTY(PL)=2THENGOSUB340:Q=10:GOTO330
                                    :rem 163
310 GETA$:IFA$<>"I"ANDA$<>"M"ANDA$<>"J"AN
    DA$<>"K"THEN310                 :rem 149
320 Q=10:D=(A$="J")*-3+(A$="K")*-1+(A$="M
    ")*-2                           :rem 183
330 IN(SI,PL)=DR(D):IN=IN(SI,PL):RETURN
                                    :rem 190
340 IFSI=15THEND=INT(RND(1)*4):RETURN
                                    :rem  47
350 IFFL=>4THEND=INT(RND(1)*4):RETURN
                                    :rem  50
360 D=INT(RND(1)*4):IF(SIAND2↑D)=2↑DTHEN3
    60                              :rem  80
370 FL=FL+1:RETURN                  :rem 113
380 SI=0:FORX=0TO3:I=PEEK(DR(X)+P)
                                    :rem  80
390 IFI<>32THENSI=SI+2↑X            :rem  10
400 NEXT:RETURN                     :rem 237
410 S1=0:S2=0:X=(P(PL)+P)/2:IFABS(X-P)=1T
    HEND=40:GOTO430                 :rem  60
420 D=1                             :rem  72
430 FORY=0TO3:Z=PEEK(X+DR(Y)+D):IF(Z=66)O
    R(Z=67)THENS1=S1+2↑Y            :rem  46
440 NEXT                            :rem 215
450 FORY=0TO3:Z=PEEK(X+DR(Y)-D):IFZ=(66)O
    R(Z=67)THENS2=S2+2↑Y            :rem  52
460 NEXT:RETURN                     :rem 243
470 GOSUB630:PRINT"{ 19 SPAZI}"     :rem 191
480 PRINT"{ 18 SPAZI}"              :rem 109
490 PRINT"{ 18 SPAZI}":GOTO520      :rem 120
500 IFPEEK(P+IN*2)=81THEN170        :rem 117
510 GOSUB630:PRINT:GOSUB480:GOSUB630:PRIN
    T"{GIU'}ERRORE MOSSA":GOSUB290:GOTO14
    0                               :rem 234
520 PRINT"{HOME}":FORX=1TO4         :rem  57
530 PRINT TAB(29);"GIOC."X;:POKE646,CL(X)
    :PRINT"Q{WHT}"                  :rem 104
540 PRINTTAB(30);B(X):NEXT          :rem  80
550 RETURN                          :rem 122
560 PRINT"{CLR}{ 6 GIU'}{RVS}[<1>]"SPC(16
    )"RETTANGOLI"                   :rem 211
570 PRINT"{ 7 GIU'}{ 10 SPAZI}{CYN}1. CON
    TROLLO GIOCATORE                :rem  25
580 PRINT"{WHT}{GIU'}{ 10 SPAZI}{GRN}2. C
    ONTROLLO COMPUTER               :rem  11
590 FORX=1TO4                       :rem  34
600 PRINT"{YEL}{HOME}{ 10 GIU'}
    { 6 SPAZI}SERPENTE"X" (SCEGLI 1 O 2)
    {WHT}"                          :rem 131
610 GETA$:IFVAL(A$)>2ORVAL(A$)=0THEN610
                                    :rem  27
620 TY(X)=VAL(A$):NEXT:GOTO80       :rem  24
630 PRINT"{HOME}":FORQQ=1TO18:PRINT:NEXT:
    RETURN                          :rem  20
640 IFQF=0THEN100                   :rem 237
650 PRINT"{CLR}{ 6 GIU'}"SPC(14)"{RVS}
    [<1>]GAME OVER!{ 3 GIU'}"       :rem 130
660 FORX=1TO4:POKE 646,CL(X):PRINTTAB(7)"
    {GIU'}GIOCAT."X;"...."B(X)" RETTANGOL
    I"                              :rem 164
670 NEXT                            :rem 220
680 PRINT"{ 3 GIU'}"SPC(10)"{WHT}ANCORA?
    (S/N)":POKE198,0                :rem 254
690 GETA$:IFA$="S"THENRUN           :rem  10
700 IFA$="N"THENPRINT"{CLR}":END:rem 254
710 GOTO690                         :rem 112
```

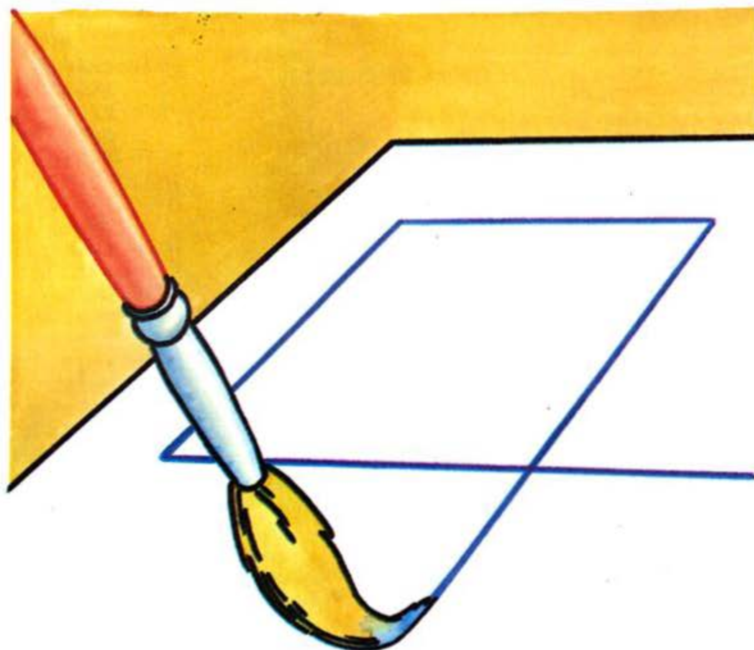




Paintbox

di **C. Metcalf**
trad. ed adatt.
di **L. Priotto**

Una delle caratteristiche più potenti del C64 sono i grafici in alta risoluzione a colori. Tuttavia, come ogni sistema potente e versatile, può risultare difficile da usare e programmare. Ecco finalmente un programma che fa diventare uno scherzo la gestione della grafica in alta risoluzione e multicolor sul vostro C64, rendendolo parzialmente compatibile con i programmi scritti per i computer Atari.



PAINTBOX implementa sul C64 i potenti comandi grafici dei computer Atari, permettendo di disporre della eccezionale grafica del C64 (superiore a quella Atari) con un set di comandi Atari (molto più potente delle PEEK e POKE richieste da C64), che consente di ottenere dei risultati eccezionali.
Con PAINTBOX in memoria possono essere digitati anche programmi scritti per computer Atari, che usino dei modi grafici 7 e 8, semplicemente facendo precedere ogni comando grafico Atari da un punto esclamativo.
PAINTBOX è diviso in due programmi: il primo contiene il linguaggio macchina di PAINTBOX e lo inserisce in memoria, provvedendo quindi a caricare ed eseguire automaticamente il programma seguente, consistente in una dimostrazione delle possibilità offerte da PAINTBOX. Se desiderate utilizzare PAINTBOX nei vostri stessi programmi, potrete cancellare dal primo programma le istruzioni non necessarie, in particolare quelle che provvedono al caricamento del secondo programma (linee 160-350); in questo caso dovrete dare il RUN al programma e digitare, alla ricomparsa del cursore, SYS49152 seguito da NEW. Così facendo il vostro C64 sarà pronto ad eseguire programmi BASIC contenenti i comandi di PAINTBOX.
I comandi grafici Atari sono semplici da usare: ogni comando, come detto, deve (solo nel C64, si noti bene!) essere preceduto da un punto esclamativo (e da due punti se il comando segue un'istruzione IF... THEN), per fare in modo che l'interprete BASIC del C64 li riconosca come comandi speciali. La sintassi del comando, al pari di tutti i comandi del C64, può essere scritta per intero oppure in forma abbreviata con un punto, proprio come sugli Atari. Queste abbreviazioni non vengono riportate comunque in forma estesa, se si lista il programma 1. I vari parametri dell'istruzione seguono l'istruzione stessa; un esempio può essere:

```
! PLOT 100,100
```

per plottare un pixel alle coordinate 100,100.
Come nei normali comandi BASIC, gli spazi vengono ignorati. Una nota importante riguarda l'istruzione IF...THEN: essa non riconoscerà come validi i comandi se non sono preceduti, oltre che dal punto esclamativo, anche dai due punti. Si immagini, ad esempio, di voler plottare un pixel in una posizione dove non ci sia già un pixel acceso. La forma dell'istruzione sarà:

```
! LOCATE10,15,A:IFA=OTHEN: !COLOR1:!PLOT10,15
```

Ecco ora un elenco dei comandi disponibili con PAINTBOX, con le relative

istruzioni per l'utilizzo. Le rispettive forme abbreviate sono riportate tra parentesi tonde.

I comandi di PAINTBOX

!GRAPHICS n (!G.) Questo comando è identico al comando Atari GRAPHICS, e richiede un solo parametro – n – che contraddistingue il modo grafico. Dal momento che PAINTBOX permette soltanto l'uso dei modi 7 e 8, n dovrà avere uno di questi due valori. Tutti i valori eventualmente assunti da n, e compresi tra 1 e 6, verranno trattati come se fossero 0. Come nell'Atari, n può essere sommato a 16, 32 o 48. n+16 non crea finestre di testo (pagina grafica totale), n+32 non pulisce la pagina grafica quando la si attiva, n+48 è una combinazione dei due precedenti.
Se non si usano questi numeri "extra", cioè se si usa per esempio !GRAPHICS8, la pagina grafica verrà pulita e una finestra per il testo, composta da quattro linee, verrà definita nella parte bassa dello schermo. A prescindere dai numeri addizionali, le schermate verranno comunque riportate ai colori standard Atari.
È importante non cercare di utilizzare il nastro o il disco, se la finestra per il testo è abilitata: per esempio, se date un LOAD e poi premete RUN/STOP, le routine di raster interrupt verranno parzialmente disabilitate e dovrete rientrare in modo grafico (usando n+32). Qualsiasi tentativo di accesso al disco darà un "?DEVIC⁼ NOT PRESENT ERROR".
Gli Atari non permettono il plottaggio di pixel nell'area "sotto" la finestra per il testo, mentre PAINTBOX ammette anche questa possibilità. In questo caso, naturalmente, la parte di grafico "sotto" la finestra per il testo rimarrà invisibile fino a che si dà un comando !GRAPHICS n+48, con n che vale 7 o 8.
Inoltre, se si lavora in modo immediato con la pagina grafica, PAINTBOX non abbisogna della finestra di testo, mentre l'Atari ha questa necessità.

!PLOT x,y (!P.) Plotta un pixel alle coordinate x,y. L'origine delle coordinate si trova nell'angolo superiore sinistro dello schermo. In modo grafico 8 i limiti per le coordinate sono 0<=x<=319 e 0<=y<=199. In modo grafico 7 questi limiti si riducono a 0<=x<=159 e 0.min<=y<=99. Questo comando non è abilitato per funzionare, se il modo grafico è posto a zero. Il plottaggio avviene nel colore contenuto nel corrente registro di colore (vedi SETCOLOR e COLOR). PLOT definisce inoltre il punto di partenza per il comando DRAWTO, plottando contemporaneamente.

!POSITION x,y (!PO.) Questo comando definisce il punto di partenza per il comando DRAWTO, senza alterare in alcun modo ciò che è visualizzato sul video.
I valori di x ed y sono gli stessi del comando PLOT.

!DRAWTO x,y (!D.) Traccia una linea a partire dal punto di arrivo del precedente comando DRAWTO, PLOT o POSITION, fino al punto definito dalle coordinate x ed y. I valori di queste ultime sono gli stessi del comando PLOT.

!SETCOLOR r,c1,c2 (!S.) Modifica il contenuto del registro colore r con il colore c1 e con la luminanza c2. c1 e c2 possono variare da 0 a 15. I registri colore definiscono i colori di bordo, schermo, testo e pixel, come definito dalla seguente tabella:

valore di r	GRAPHICS 0	GRAPHICS 7	GRAPHICS 8
0	–	coppie pixel 01	–
1	caratteri	coppie pixel 10	caratteri/pixel
2	fondo	coppie pixel 11	fondo
3	–	–	–
4	bordo	fondo	bordo

Si noti che le coppie di pixel (00, 01,10 e 11) sono usate per definire i singoli pixel in GRAPHICS 7, esattamente come in modo multicolor nel C64.
Uno spiacevole problema, dipendente dalla diversa configurazione del C64 e dell'Atari, è che in modo grafico 7 il colore dei caratteri nella finestra testo del C64 è definito dal registro SETCOLOR numero 2, e non dal numero 1. In questo modo la finestra testo non può avere un proprio colore indipendente, ma assume il colore del resto dello schermo.
Inoltre, un altro problema con il registro 2 nel modo grafico 7 è che quel registro viene posto sul colore del fondo (nei modelli più vecchi di C64 è il bianco) ogni volta che lo schermo viene pulito. In questo modo la stampa del classico cuoricino inverso mentre ci si trova in modo grafico 7 deve essere evitata: tutte le coppie di pixel 11 diventerebbero del colore del fondo, cioè invisibili. Oltre a ciò, ogni scroll del testo in modo grafico 7 provocherà uno scroll di dati scorretti nelle coppie di pixel 11. Nessun problema invece nel modo grafico 8.
È interessante sapere che l'esecuzione di una !SETCOLOR 2, c1, c2 in GRAPHICS 7, o di una !SETCOLOR 1, c1, c2 in GRAPHICS 8, fa in modo che il registro colore caratteri nella locazione 646 venga regolato sui colori c1,c2. Quindi i precedenti codici colore non vengono considerati quando viene eseguito un comando !SETCOLOR o !GRAPHICS (infatti !GRAPHICS chiama automaticamente !SETCOLOR per regolare i propri colori di default).
I numeri da 0 a 15 che possono essere usati in SETCOLOR corrispondono esattamente ai vari colori e relative luminanze disponibili sugli Atari.

!COLOR r (!C.) Specifica quale registro colore (visti prima per SETCOLOR) deve essere usato per il plottaggio di punti o linee. In entrambi i modi grafici un valore zero provoca la cancellazione dei pixel. In GRAPHICS8 un numero dispari per r regola sempre il computer sul plottaggio dei pixel. In GRAPHICS7 vengono usati i registri 1–3, dove il registro 1 regola il colore delle coppie di pixel 01, il 2 delle 10 e il 3 delle 11 (si noti che esso è il numero di SETCOLOR più 1).

!LOCATE x,y,v (!L.) Questo comando restituisce (nella variabile v a virgola mobile) il pixel corrente alle coordinate x,y e definisce il punto di partenza per una DRAWTO al pixel letto dalla LOCATE. Quindi, in GRAPHICS 8 viene restituito nella variabile v o uno 0 (nessun pixel) oppure un 1 (pixel acceso).
In modo grafico 7 uno zero indica nessun pixel, mentre valori da 1 a 3 corrispondono rispettivamente alle coppie di pixel 01,10 e 11.

!FILL x,y (!F.) Questo comando è ancora più potente di quello disponibile sugli Atari. Colora qualsiasi area, indipendentemente dalla sua forma. Si fermerà incontrando qualsiasi pixel acceso, oppure i confini dello schermo. I parametri x ed y determinano il punto di inizio del FILL e definiscono inoltre il punto di partenza del successivo comando DRAWTO.

!TEXT x,y,"stringa" (!T.) Permette di inserire del testo a partire da qualsiasi riga e colonna in modo grafico 8. Funziona anche in modo grafico 7, ma ovviamente – trattandosi di multicolor – il testo risulta pochissimo leggibile.
Il parametro "stringa" può essere composto sia da una stringa tra virgolette

che da una variabile di stringa o una combinazione delle due. Può inoltre essere passato un parametro aggiuntivo prima di "stringa": uno 0 o un 1 in questa posizione determina se il computer deve usare il set minuscolo/maiuscolo oppure il set maiuscolo/grafico. Il programma viene inizializzato per usare il set minuscolo/maiuscolo. Nessun carattere di controllo verrà stampato, mentre i RVS ON e RVS OFF avranno il loro effetto abituale. Si ricordi che i parametri x e y devono essere specificati per ogni comando TEXT.

!QUIT (!Q.) Disabilita PAINTBOX. Si può rientrare in PAINTBOX digitando SYS49152.

Note tecniche

Le locazioni 3 e 4 contengono due variabili usate dall'interrupt che crea la finestra per il testo, per stabilire il set maiuscolo/grafico per la finestra e alta risoluzione/multicolor per il resto dello schermo. Se si desidera usare la locazione 3 per controllare il set di caratteri nella finestra, occorre eseguire POKE3,21 per avere il set maiuscolo/grafico e POKE3,23 per il set minuscolo/maiuscolo (si noti che il set minuscolo è richiesto per inserire i comandi in modo minuscolo/maiuscolo).
Il registro 4 è usato dal programma per determinare i pixel da plottare, il valore restituito da LOCATE e così via... Può essere inoltre usato per passare dall'alta risoluzione standard (POKE4,8) all'alta risoluzione in multicolor (POKE4,24). Altri valori possono generare effetti interessanti.
Ecco ora la configurazione di memoria assunta al C64 quando si utilizza PAINTBOX:



Locazioni	Commento
$0400–$07E7	finestra per il testo
$0800–$9FFF	spazio per i programmi BASIC
$A000–$BC7F	ROM del BASIC con RAM sottostante
$BC80–$BFFF	tavole dati di PAINTBOX e stack routine di FILL
$C000–$C7FF	codice macchina di PAINTBOX
$C800–$CBFF	memoria colore di PAINTBOX usata in tutti i casi tranne che per i pixel – 11 in modo grafico 8
$CC00–$CFFF	libero per utilizzi in codice macchina (es. DOS Wedge)
$E000–$FFFF	ROM Sistema Operativo, con pagina grafica nella RAM sottostante.

Immagazzinamento variabili:
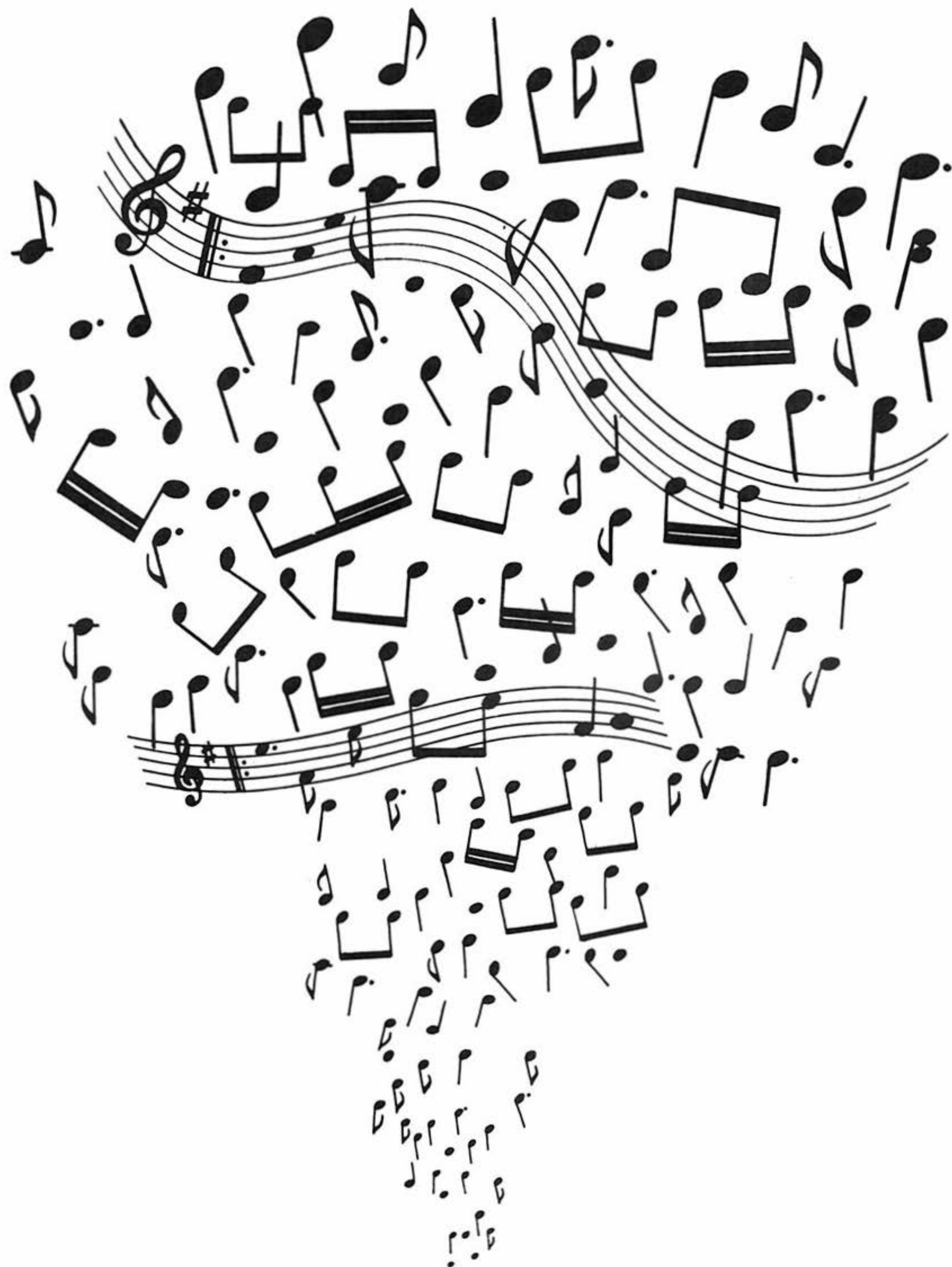
PERMANENTE: locazioni 3-6, 251-254 (immagini di interrupt 3=53272, 4=53270)
TEMPORANEA: locazioni 27-42, 107-113, 158-159, 163-164, 167-170
NON PAGINA ZERO: locazioni 670-699

PAINTBOX

```
100 POKE53280,0:POKE53281,0:POKE646,1
                                   :rem  76
110 PRINT"{CLR}MEMORIZZAZIONE LINGUAGGIO
    MACCHINA..."                   :rem 221
120 FORI=49152TO51197:READD:CK=CK+D:POKEI
    ,D                             :rem 109
130 PRINT"{HOME}{ 2 GIU'}CHECKSUM:"CK:NEX
    T                              :rem  46
140 IFCK<>251725THENPRINT"ERRORE NELLE IS
    TRUZIONI DATA!":END            :rem  13
150 PRINT"{GIU'}DATI OK. MEMORIZZAZIONE T
    ERMINATA"                      :rem  74
160 PRINT"CARICAMENTO DEL PROGRAMMA DIMOS
    TRATIVO."                      :rem 220
170 PRINT"SE SI DESIDERA USARE PAINTBOX C
    ON I"                          :rem 236
180 PRINT"PROPRI PROGRAMMI BASIC, ELIMINA
    RE DA"                         :rem 157
190 PRINT"QUESTO PROGRAMMA LE LINEE DA 30
    0 A 400"                       :rem 221
200 PRINT"{GIU'}PREMERE UN TASTO"
                                   :rem 178
210 GETA$:IFA$=""THEN210           :rem  73
300 SYS49152:PRINT"{CLR}NEW"       :rem  31
350 POKE198,3:POKE631,19:POKE632,13:POKE6
    33,131                         :rem 187
400 END                            :rem 107
1000 DATA 169,54,133,1,169,224,141,160
                                   :rem  69
1010 DATA 188,169,0,141,128,188,170,189
                                   :rem 141
1020 DATA 128,188,24,105,64,157,129,188
                                   :rem 141
1030 DATA 189,160,188,105,1,157,161,188
                                   :rem 137
1040 DATA 232,224,24,144,234,169,1,160
                                   :rem  66
```

```
1050 DATA 7,153,199,188,153,192,188,10
                              :rem  95
1060 DATA 153,207,188,136,153,192,188,10
                              :rem 185
1070 DATA 136,16,238,169,3,160,6,153
                              :rem 237
1080 DATA 216,188,10,10,136,136,16,247
                              :rem  75
1090 DATA 169,254,160,7,153,224,188,56
                              :rem  94
1100 DATA 42,136,16,248,169,252,160,7
                              :rem  28
1110 DATA 153,231,188,153,239,188,153,247
                              :rem 239
1120 DATA 188,56,42,56,42,136,136,16
                              :rem 240
1130 DATA 239,169,66,141,0,3,169,197
                              :rem 247
1140 DATA 141,1,3,169,134,141,8,3
                              :rem  71
1150 DATA 169,192,141,9,3,169,8,133
                              :rem 194
1160 DATA 4,169,55,133,1,96,160,1
                              :rem  83
1170 DATA 177,122,201,33,240,3,76,228
                              :rem  22
1180 DATA 167,165,212,208,249,32,115,0
                              :rem  76
1190 DATA 165,122,133,158,165,123,133,159
                              :rem 231
1200 DATA 162,255,160,0,165,158,133,122
                              :rem 115
1210 DATA 165,159,133,123,232,32,115,0
                              :rem  61
1220 DATA 41,127,221,242,192,240,245,201
                              :rem 159
1230 DATA 46,240,26,9,128,221,242,192
                              :rem  25
1240 DATA 240,19,189,242,192,48,3,232
                              :rem  33
1250 DATA 208,248,200,200,224,53,144,212
                              :rem 160
1260 DATA 162,11,76,66,197,185,40,193
                              :rem  43
1270 DATA 141,233,192,185,41,193,141,234
                              :rem 177
1280 DATA 192,32,115,0,169,54,133,1
                              :rem 178
1290 DATA 32,46,194,169,55,133,1,76
                              :rem 198
1300 DATA 174,167,68,82,65,87,164,80
                              :rem   1
1310 DATA 76,79,212,80,79,83,73,84
                              :rem 158
1320 DATA 73,79,206,76,79,67,65,84
                              :rem 167
1330 DATA 197,83,69,84,67,79,76,176
                              :rem 228
1340 DATA 67,79,76,176,71,82,65,80
                              :rem 163
1350 DATA 72,73,67,211,70,73,76,204
                              :rem 191
1360 DATA 81,85,73,212,84,69,88,212
                              :rem 205
1370 DATA 138,194,46,194,31,194,181,196
                              :rem 151
1380 DATA 199,195,150,196,81,193,242,197
                              :rem 209
1390 DATA 60,193,252,198,169,228,141,8
                              :rem  99
1400 DATA 3,169,167,141,9,3,169,139
                              :rem 195
1410 DATA 141,0,3,169,227,141,1,3
                              :rem  66
1420 DATA 96,32,42,197,208,39,138,48
                              :rem 254
1430 DATA 36,41,15,168,192,7,176,32
                              :rem 191
1440 DATA 120,32,0,194,88,169,27,141
                              :rem 236
1450 DATA 17,208,169,23,141,24,208,169
                              :rem  85
1460 DATA 8,141,22,208,133,4,169,199
                              :rem 243
1470 DATA 141,0,221,208,102,76,61,197
                              :rem  23
1480 DATA 192,9,176,249,120,169,59,141
                              :rem 100
1490 DATA 17,208,169,40,141,24,208,169
                              :rem  88
1500 DATA 196,141,0,221,169,8,192,7
                              :rem 186
1510 DATA 208,2,169,24,133,4,141,22
                              :rem 172
1520 DATA 208,169,23,133,3,138,41,16
                              :rem 231
1530 DATA 208,35,169,127,141,13,220,169
                              :rem 129
1540 DATA 1,141,26,208,141,18,208,169
                              :rem  26
1550 DATA 198,141,38,3,169,197,141,39
                              :rem  50
1560 DATA 3,169,100,141,20,3,169,197
                              :rem 234
1570 DATA 141,21,3,208,3,32,0,194
                              :rem  71
1580 DATA 88,138,41,32,208,18,160,0
                              :rem 188
1590 DATA 132,168,169,0,133,170,162,224
                              :rem 130
1600 DATA 32,93,196,169,147,32,210,255
                              :rem  86
1610 DATA 169,4,133,158,166,158,188,251
                              :rem 147
1620 DATA 193,132,168,32,8,196,198,158
                              :rem 102
1630 DATA 16,242,96,8,14,6,9,0  :rem 194
1640 DATA 169,0,141,26,208,169,129,141
                              :rem  83
1650 DATA 13,220,169,202,141,38,3,169
                              :rem  26
1660 DATA 241,141,39,3,169,49,141,20
                              :rem 236
1670 DATA 3,169,234,141,21,3,96,32
                              :rem 136
1680 DATA 228,196,160,2,185,167,2,153
                              :rem  41
1690 DATA 251,0,136,16,247,96,32,31
                              :rem 188
1700 DATA 194,32,24,197,240,7,230,253
                              :rem  28
1710 DATA 32,61,194,198,253,32,66,194
                              :rem  44
1720 DATA 240,45,165,253,74,74,74,170
                              :rem  38
1730 DATA 165,251,69,253,41,248,69,253
                              :rem  98
```

```
1740 DATA 24,125,128,188,133,195,189,160 :rem 195
1750 DATA 188,101,252,133,196,165,251,41 :rem 183
1760 DATA 7,32,24,197,240,5,41,254 :rem 138
1770 DATA 13,170,2,170,160,0,96,169 :rem 185
1780 DATA 53,120,133,1,177,195,160,54 :rem 31
1790 DATA 132,1,88,61,224,188,164,254 :rem 44
1800 DATA 240,5,29,192,188,160,0,145 :rem 235
1810 DATA 195,96,32,228,196,173,167,2 :rem 50
1820 DATA 56,229,251,141,180,2,173,168 :rem 84
1830 DATA 2,229,252,141,181,2,173,169 :rem 31
1840 DATA 2,56,229,253,133,107,160,1 :rem 230
1850 DATA 162,0,32,24,197,240,1,200 :rem 170
1860 DATA 165,252,205,168,2,144,36,208 :rem 86
1870 DATA 7,173,167,2,197,251,176,27 :rem 0
1880 DATA 160,255,162,255,32,24,197,240 :rem 137
1890 DATA 1,136,165,251,56,237,167,2 :rem 246
1900 DATA 141,180,2,165,252,237,168,2 :rem 27
1910 DATA 141,181,2,132,111,134,112,160 :rem 103
1920 DATA 1,32,24,197,240,1,200,173 :rem 171
1930 DATA 169,2,197,253,176,15,152,73 :rem 46
1940 DATA 255,24,105,1,168,165,253,56 :rem 37
1950 DATA 237,169,2,133,107,132,167,169 :rem 142
1960 DATA 0,141,182,2,133,163,174,180 :rem 23
1970 DATA 2,172,181,2,208,14,228,107 :rem 234
1980 DATA 176,10,166,107,32,37,195,133 :rem 89
1990 DATA 163,76,46,195,32,37,195,141 :rem 53
2000 DATA 182,2,76,46,195,132,110,152 :rem 19
2010 DATA 74,134,109,138,106,96,169,0 :rem 32
2020 DATA 133,158,133,159,133,164,141,183 :rem 225
2030 DATA 2,32,49,194,165,252,205,168 :rem 32
2040 DATA 2,208,17,165,251,205,167,2 :rem 227
2050 DATA 208,10,165,253,205,169,2,208 :rem 73
2060 DATA 3,76,34,194,165,163,24,109 :rem 241
2070 DATA 180,2,133,163,165,164,109,181 :rem 126
2080 DATA 2,133,164,197,110,240,4,144 :rem 18
2090 DATA 33,176,6,165,163,197,109,144 :rem 93
2100 DATA 25,165,163,229,109,133,163,165 :rem 178
2110 DATA 164,229,110,133,164,165,251,24 :rem 168
2120 DATA 101,111,133,251,165,252,101,112 :rem 194
2130 DATA 133,252,173,182,2,24,101,107 :rem 60
2140 DATA 141,182,2,173,183,2,105,0 :rem 167
2150 DATA 141,183,2,197,110,240,4,144 :rem 16
2160 DATA 32,208,7,173,182,2,197,109 :rem 240
2170 DATA 144,23,173,182,2,229,109,141 :rem 75
2180 DATA 182,2,173,183,2,229,110,141 :rem 20
2190 DATA 183,2,165,253,24,101,167,133 :rem 75
2200 DATA 253,76,57,195,76,61,197,32 :rem 254
2210 DATA 42,197,208,248,224,5,176,244 :rem 87
2220 DATA 138,72,32,35,197,138,41,15 :rem 238
2230 DATA 10,10,133,168,32,35,197,138 :rem 23
2240 DATA 41,15,74,170,240,3,74,5 :rem 77
2250 DATA 168,133,168,74,168,185,118,196 :rem 206
2260 DATA 176,4,74,74,74,74,41,15 :rem 97
2270 DATA 164,168,192,3,208,6,224,7 :rem 193
2280 DATA 208,2,169,1,133,168,104,170 :rem 26
2290 DATA 224,3,240,36,160,240,165,168 :rem 77
2300 DATA 32,24,197,208,28,224,0,240 :rem 225
2310 DATA 23,202,208,5,32,82,196,240 :rem 225
2320 DATA 31,224,1,208,5,32,45,196 :rem 127
2330 DATA 240,32,202,202,202,157,32,208 :rem 105
2340 DATA 96,202,48,12,202,48,19,240 :rem 233
2350 DATA 25,202,32,45,196,202,240,237 :rem 71
2360 DATA 160,15,165,168,10,10,10,10 :rem 213
2370 DATA 133,168,169,204,133,170,162,200 :rem 224
2380 DATA 208,11,162,216,169,220,133,170 :rem 172
2390 DATA 165,168,141,134,2,132,6,160 :rem 26
2400 DATA 0,132,195,134,196,177,195,37 :rem 89
2410 DATA 6,5,168,145,195,200,208,245 :rem 31
```



A TUTTA MUSICA CON SOUND

Musica dal calcolatore, musica vera, completa, definitiva: da ascoltare con gli amici. Ed è musica tua!

Con l'unità periferica SOUND BUGGY e la tastierina musicale SIEL da appoggiare sopra la tastiera alfanumerica, il tuo Commodore 64 si trasforma in una autentica band e tu diventi subito concertista, compositore e arrangiatore.

Musicista in pochi giorni

Se sei già esperto di musica, SOUND BUGGY ti porterà alla perfezione. Se sei principiante, ti troverai in pochi giorni di fronte a un miracolo: comporrai musica tua e potrai ascoltarla in una perfetta registrazione elettronica collegabile a ogni impianto stereo, videotelevisivo, monitor C64.

Un pacchetto di programmi che concentra anni di studio

Grazie al consistente, straordinario software di Sound Buggy, potrai eseguire o comporre su 24 ritmi (12 preregistrati), disponendo di ben 28 timbri strumentali (14 preregistrati), potendo correggere, migliorare, variare e disporre quindi per la registrazione definitiva.

Tramite interfaccia MIDI, SOUND BUGGY comunica anche con gli altri strumenti musicali elettronici (expander, sintetizzatori, sequencer ecc.). SOUND BUGGY è un prodigio dell'elettronica al servizio della creatività.

SIEL
POWER
RHYTHM
ACCOMP.
MASTER
SOU

Spia luminosa di corretta alimentazione

Volume per la batteria elettronica

Volume per la parte orchestrale

Volume generale



MAXIPRESTAZIONE IN MINISPAZIO

Il tuo laboratorio musicale, completo e perfetto, è tutto qui: il Commodore 64, il SOUND BUGGY, la minitastiera SIEL, il software.

SIEL®
LA QUALITÀ DELLA MUSICA

BUGGY

Main Menu
Massima semplicità d'impiego del software grazie alla standardizzazione dei comandi. Il Main Menu presenta il quadro complessivo delle funzioni musicali utilizzabili premendo i comandi indicati.







Rhythm Menu
Visualizza tutti i 24 ritmi e mostra come SOUND BUGGY li esegue alla batteria elettronica. La batteria è programmabile per: Bass Drum (grancassa), Cymbal Short (piatto aperto) Cymbal Long (piatto chiuso), Snare Drum (tamburo rullante) e Rim Shot (cerchio rullante).

Solo Menu
Elenca tutti i 28 timbri e visualizza i parametri con cui il sint SOUND BUGGY li sviluppa.

Song Edit
È il cuore del SOUND BUGGY e permette di dirigere l'"orchestra". Infatti consente di intervenire in tempo reale, modificando i parametri relativi a: Solo, Rhythm, Accompaniment, Sequencer.

Compila e spedisci l'unita Cedola Privilegiata. È l'unico mezzo per ricevere SOUND BUGGY, completo di tutti i programmi, su disco e cassetta, a un prezzo speciale, direttamente a domicilio e con 12 mesi di Garanzia.

AFFRETTATI!

Il prezzo è bloccato solo fino al 31-12-'85.

CEDOLA PRIVILEGIATA DI ACQUISTO
SOUND BUGGY

Indicare con una X il ☐ che interessa



SIEL

☐ Sì desidero acquistare SOUND BUGGY, la vostra unità periferica per C64. Speditemela contrassegno completa del pacchetto software sia su disco che su cassetta e libretto istruzioni al prezzo speciale di Lire 185.000 (incluse L. 27.650 IVA e L. 3.750 di spese postali).
È inteso che il mio SOUND BUGGY sarà coperto da Garanzia per 1 anno.

☐ *Sono interessato alla unità periferica per C64 SOUND BUGGY e vi prego di volermi fornire informazioni più dettagliate.*

12 MESI DI GARANZIA

NOME ______ COGNOME ______

INDIRIZZO ______

CAP ______ LOCALITÀ ______

DATA ______ FIRMA ______
(per i minori occorre quella del genitore)

DATI FACOLTATIVI

ETÀ ______ PROFESSIONE ______

TITOLO DI STUDIO ______

SOUND  BUGGY

TUTTA LA MUSICA A COMANDO!

A tua disposizione hai gli strumenti, i ritmi, i tempi, gli accordi, gli arpeggi, gli effetti mixati, come una grande, insperata tavolozza di suoni. Ma tutto questo resterebbe muto senza l'intervento della tua creatività: sotto la guida del tuo estro musicale, seguendo le facilissime istruzioni dei programmi, prenderà vita sotto le tue mani dapprima la musica che vuoi produrre, poi la tua musica.

Sequencer

È una funzione polifonica che ti permette di comporre e registrare in tempo reale fino a 9 song in linea, archiviarle su disco e creare così una discoteca da richiamare a tuo piacere.

Rhythm



Un'intera batteria elettronica al tuo comando con tre parametri modificabili: Start, Tempo, Key-Start. C'è anche il Down Beat che, come un metronomo, visualizza le battute musicali.

Accordi

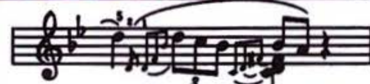


Grazie a due sofisticate tecniche di sviluppo, basta un dito per esplorare tutte le possibilità armoniche di base e creare accordi automatici.

Orchestrazioni



Creato l'accordo il SOUND BUGGY si fa in quattro: basso, batteria, arrangiamento con arpeggio e accompagnamento ritmato trasformandoti in direttore d'orchestra.



Ha anche una grande tastiera.

Con la piccola tastiera sovrapposta a quella del computer, SOUND BUGGY ti dà tutto ciò che puoi chiedere al tuo gusto musicale. Ma con la grande tastiera SIEL CMK 49 a passo professionale, l'elettronica ti offre una ricchezza musicale assoluta, con la possibilità di impostare la tua musica a due mani su un'unica tastiera, come nel classico pianoforte.



Con CMK 49 puoi diventare un vero tastierista.

Qualsiasi altra tastiera collegabile al tuo Commodore 64 diventa "magica" con SOUND BUGGY: perchè il suo software è universale.

MIDI

(Musical Instruments Digital Interface)



Grazie a questo linguaggio comune tra sintetizzatori professionali, SOUND BUGGY è in grado di parlare con 4 tastiere professionali o expander creando delle orchestrazioni di eccezionale qualità timbrica. L'importanza di questo fattore è ben conosciuta dal musicista esperto; per chi lo vuol diventare SOUND BUGGY sarà una meravigliosa scoperta.

AVVISO IMPORTANTE

Il SOUND BUGGY è acquistabile esclusivamente tramite il coupon privilegiato di questa offerta. Tutti gli altri strumenti elettronici SIEL sono in vendita presso i migliori Negozi di strumenti musicali. Se siete interessati rivolgetevi a loro. Oppure, se desiderate altre informazioni, scrivete a "Filodiretto SIEL" Casella Postale 10823 20124 Milano MI.

COUPON GARANZIA 12 MESI

Compila e spedisci in busta chiusa a:

"Filodiretto SIEL"

SIEL®

SOCIETÀ INDUSTRIE ELETTRONICHE s.p.a.

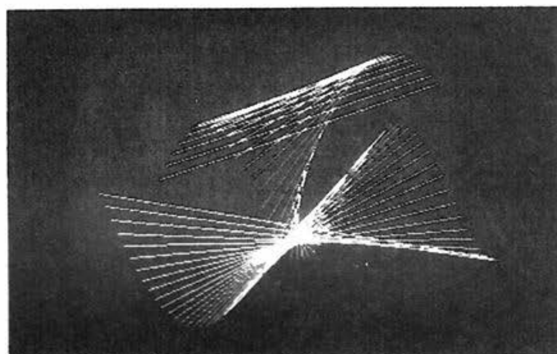
CASELLA POSTALE 10823
20124 Milano MI

SIEL®

SIEL S.p.a.
Stampa propagandistica Semestrale n. 1. II semestre 1985. Sped. Abb. Postale V Gruppo. Vendite per Corrispondenza. Proposta, accettazione, prezzo e invio del prodotto si realizzeranno attraverso i Servizi Postali. C.C.P. n. 12829636.

```
2420 DATA 232,228,170,208,238,96,11,207
                                :rem 130
2430 DATA 199,113,40,143,146,40,153,170
                                :rem 123
2440 DATA 102,68,102,68,102,238,102,238
                                :rem 122
2450 DATA 100,227,110,227,110,93,85,93
                                :rem  73
2460 DATA 85,215,136,93,136,170,32,42
                                :rem  34
2470 DATA 197,138,41,3,32,24,197,208
                                :rem 246
2480 DATA 5,41,1,133,254,96,133,254
                                :rem 186
2490 DATA 201,0,208,2,169,1,10,10
                                :rem  63
2500 DATA 10,141,170,2,96,32,31,194
                                :rem 172
2510 DATA 32,234,198,32,170,198,72,169
                                :rem  94
2520 DATA 55,133,1,32,115,0,32,139
                                :rem 119
2530 DATA 176,32,133,177,104,168,169,0
                                :rem  85
2540 DATA 32,145,179,165,98,41,127,133
                                :rem  92
2550 DATA 98,160,4,185,97,0,145,71
                                :rem 150
2560 DATA 136,16,248,96,32,42,197,32
                                :rem 250
2570 DATA 12,197,152,240,8,192,2,176
                                :rem 242
2580 DATA 76,224,64,176,72,142,167,2
                                :rem 250
2590 DATA 140,168,2,32,35,197,32,12
                                :rem 186
2600 DATA 197,152,208,57,224,200,176,53
                                :rem 133
2610 DATA 142,169,2,96,32,24,197,240
                                :rem 241
2620 DATA 6,138,10,170,152,42,168,96
                                :rem 239
2630 DATA 133,170,165,4,41,16,8,165
                                :rem 184
2640 DATA 170,40,96,169,55,133,1,32
                                :rem 188
2650 DATA 253,174,169,55,133,1,32,158
                                :rem  38
2660 DATA 173,32,247,183,169,54,133,1
                                :rem  38
2670 DATA 166,20,164,21,96,162,246,154
                                :rem  88
2680 DATA 162,14,224,128,176,27,134,163
                                :rem 136
2690 DATA 72,169,55,133,1,174,21,3
                                :rem 140
2700 DATA 224,197,240,10,169,32,44,17
                                :rem  28
2710 DATA 208,240,3,32,96,193,104,166
                                :rem  30
2720 DATA 163,76,139,227,173,25,208,141
                                :rem 138
2730 DATA 25,208,169,27,141,17,208,169
                                :rem  92
2740 DATA 199,141,0,221,169,23,141,24
                                :rem  26
2750 DATA 208,169,8,141,22,208,162,0
                                :rem 235
2760 DATA 173,18,208,48,22,162,218,169
                                :rem  95
2770 DATA 196,141,0,221,169,59,141,17
                                :rem  37
2780 DATA 208,169,40,141,24,208,169,8
                                :rem  43
2790 DATA 141,22,208,142,18,208,173,13
                                :rem  77
2800 DATA 220,41,1,240,3,76,49,234
                                :rem 126
2810 DATA 56,32,240,255,224,21,176,6
                                :rem 234
2820 DATA 162,21,24,32,240,255,165,3
                                :rem 226
2830 DATA 141,117,197,165,4,141,151,197
                                :rem 137
2840 DATA 104,168,104,170,104,64,72,41
                                :rem  74
2850 DATA 127,201,32,144,4,104,76,202
                                :rem  18
2860 DATA 241,104,32,202,241,8,133,170
                                :rem  65
2870 DATA 134,158,132,159,56,32,240,255
                                :rem 140
2880 DATA 224,21,176,6,162,21,24,32
                                :rem 182
2890 DATA 240,255,166,158,164,159,165,170
                                :rem 252
2900 DATA 40,96,32,31,194,32,234,198
                                :rem 243
2910 DATA 169,0,141,174,2,169,0,141
                                :rem 180
2920 DATA 176,2,141,175,2,165,252,208
                                :rem  31
2930 DATA 4,165,251,240,33,165,251,56
                                :rem  31
2940 DATA 237,177,2,133,251,165,252,233
                                :rem 134
2950 DATA 0,133,252,32,170,198,240,229
                                :rem  80
2960 DATA 165,251,24,109,177,2,133,251
                                :rem  83
2970 DATA 165,252,105,0,133,252,230,253
                                :rem 122
2980 DATA 32,170,198,208,11,173,176,2
                                :rem  39
2990 DATA 208,11,32,212,198,169,1,44
                                :rem 243
3000 DATA 169,0,141,176,2,198,253,198
                                :rem  37
3010 DATA 253,32,170,198,208,11,173,175
                                :rem 127
3020 DATA 2,208,11,32,212,198,169,1
                                :rem 174
3030 DATA 44,169,0,141,175,2,230,253
                                :rem 224
3040 DATA 32,61,194,165,251,24,109,177
                                :rem  82
3050 DATA 2,133,251,165,252,105,0,133
                                :rem   9
3060 DATA 252,165,197,201,63,240,48,165
                                :rem 135
3070 DATA 252,240,6,165,251,201,64,176
                                :rem  76
3080 DATA 5,32,170,198,240,168,172,174
                                :rem  88
3090 DATA 2,240,28,136,185,0,189,133
                                :rem 237
```

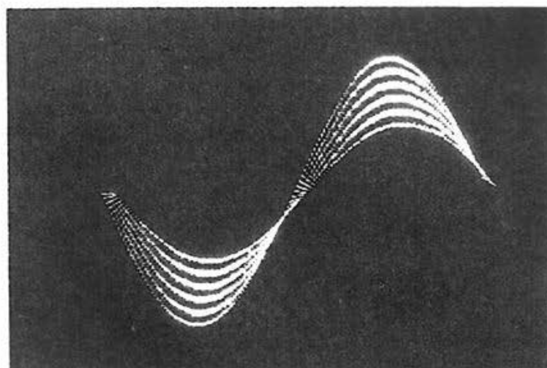
```
3100 DATA 253,185,0,190,133,252,185,0
                                :rem  18
3110 DATA 191,133,251,140,174,2,165,253
                                :rem 118
3120 DATA 201,200,176,226,76,253,197,76
                                :rem 134
3130 DATA 34,194,32,66,194,134,170,189
                                :rem  91
3140 DATA 224,188,73,255,162,53,120,134
                                :rem 129
3150 DATA 1,49,195,230,1,88,72,165
                                :rem 143
3160 DATA 170,41,7,170,104,236,178,2
                                :rem 230
3170 DATA 176,7,74,232,236,178,2,144
                                :rem 246
3180 DATA 249,201,0,96,172,174,2,165
                                :rem 239
3190 DATA 251,153,0,191,165,252,153,0
                                :rem  21
3200 DATA 190,165,253,153,0,189,238,174
                                :rem 135
3210 DATA 2,96,162,1,160,7,32,24:rem  24
3220 DATA 197,240,2,232,136,142,177,2
                                :rem  23
3230 DATA 140,178,2,96,32,42,197,208
                                :rem 242
3240 DATA 15,224,40,176,11,134,163,32
                                :rem  16
3250 DATA 35,197,208,4,224,25,144,5
                                :rem 188
3260 DATA 162,14,76,66,197,169,0,133
                                :rem 248
3270 DATA 196,165,163,10,10,10,38,196
                                :rem  30
3280 DATA 24,125,128,188,133,195,165,196
                                :rem 199
3290 DATA 125,160,188,133,196,169,55,133
                                :rem 196
3300 DATA 1,32,115,0,32,158,173,165
                                :rem 171
3310 DATA 13,48,25,32,247,183,165,20
                                :rem 232
3320 DATA 41,1,8,173,160,199,40,208
                                :rem 183
3330 DATA 3,41,247,44,9,8,141,160
                                :rem  82
3340 DATA 199,76,49,199,165,98,208,15
                                :rem  69
3350 DATA 32,133,177,160,2,177,71,153
                                :rem  29
3360 DATA 97,0,136,16,248,48,11,165
                                :rem 195
3370 DATA 23,133,22,165,23,56,233,3
                                :rem 178
3380 DATA 133,23,165,97,240,89,169,0
                                :rem 251
3390 DATA 141,180,2,173,160,199,41,251
                                :rem  80
3400 DATA 141,160,199,169,0,133,159,172
                                :rem 133
3410 DATA 180,2,177,98,32,208,199,144
                                :rem  44
3420 DATA 52,10,38,159,10,38,159,10
                                :rem 181
3430 DATA 38,159,133,158,165,159,24,105
                                :rem 142
3440 DATA 216,133,159,160,7,162,55,169
                                :rem  89
3450 DATA 51,120,133,1,177,158,145,195
                                :rem  81
3460 DATA 136,16,249,134,1,88,165,195
                                :rem  47
3470 DATA 24,105,8,133,195,144,6,230
                                :rem 234
3480 DATA 196,165,196,240,10,238,180,2
                                :rem  88
3490 DATA 173,180,2,197,97,208,180,96
                                :rem  55
3500 DATA 170,201,18,208,8,173,160,199
                                :rem  81
3510 DATA 9,4,141,160,199,201,146,208
                                :rem  28
3520 DATA 8,173,160,199,41,251,141,160
                                :rem  79
3530 DATA 199,138,41,127,201,32,144,10
                                :rem  73
3540 DATA 138,201,128,41,191,144,2,233
                                :rem  71
3550 DATA 64,56,96,13,13,0      :rem 255
```



Programma dimostrativo per PAINTBOX

```
100 REM                             :rem 117
110 REM DEMO PER 64 PAINTBOX        :rem  81
130 REM                             :rem 120
140 GOSUB700                        :rem 172
150 DATA "{WHT}ESEMPIO NUMERO 1":rem 146
160 DATA "VERRANNO DIMOSTRATE LE POSSIBIL
    ITA'"                           :rem  87
170 DATA "GRAFICHE OFFERTE DA {RVS}PAINT
    BOX{OFF}"                       :rem 200
180 DATA "IN ALTA RISOLUZIONE E MULTICOLO
    R...",                          :rem  24
190 FORI=0TO270STEP5:!PL.I,100+SIN(I/50)*
    100:!DR.319-I,100+COS(I/25)*50:NEXT
                                    :rem  68
200 FORI=1TO1000:GETA$:IFA$=""THENNEXT
                                    :rem  48
210 GOSUB700                        :rem 170
220 DATA "LA FIGURA SEGUENTE E'TRACCIATA
    PRIMA"                          :rem  14
```

```
230 DATA "IN ALTA RISOLUZIONE E POI IN MU
    LTICOLOR"                    :rem 221
240 DATA "CON EFFETTI MOLTO INTERESSANTI.
    ..",                         :rem 181
250 FORI=0TO309STEP2:!PL.I,100+SIN(I/50)*
    100:!DR.I+10,100+SIN(I/50)*50:NEXT
                                 :rem   6
260 GOSUB640:GOSUB700            :rem   3
270 DATA "FIGURA HIRES/MULTICOLOR NUMERO
    2",                          :rem 157
280 FORI=0TO309STEP2:!PL.I,100+COS(I/50)*
    100:!DR.I+10,100+SIN(I/50)*50:NEXT
                                 :rem   4
290 GOSUB640:GOSUB700            :rem   6
300 DATA "ESEMPIO NUMERO 2",     :rem 183
310 FORI=0TO319STEP2:!PL.I,100+SIN(I/50)*
    100:!DR.319-I,100+COS(I/50)*50:NEXT
                                 :rem  61
330 FORI=1TO1000:GETA$:IFA$=""THENNEXT
                                 :rem  52
340 GOSUB700                     :rem 174
350 DATA "ESEMPIO NUMERO 3",     :rem 189
390 FORI=0TO310STEP5:!PL.I,100+SIN(I/50)*
    100:!DR.319-I,100+SIN(I/50)*50:NEXT
                                 :rem  68
420 FORI=1TO1000:GETA$:IFA$=""THENNEXT
                                 :rem  52
430 GOSUB700                     :rem 174
440 DATA "LA PROSSIMA IMMAGINE E' UN CERC
    HIO."                        :rem 114
450 DATA "DURANTE LA VISUALIZZAZIONE, PRE
    MI"                          :rem 215
455 DATA "UN TASTO PER PROCEDERE NEL DEMO
    .",                          :rem   8
460 FORI=0TO2*π-π/100STEPπ/100:!PL.160
    ,100:!DR.160+COS(I)*100,100-SIN(I)*80
                                 :rem 176
470 NEXT:C=0:I=2                 :rem 182
480 !SE.1,C,I:I=I+1:IFI=16THENI=2:C=C+1:I
    FC=16THENC=0                 :rem 130
490 GETA$:IFA$=""THENFORII=1TO10:NEXT:GOT
    O480                         :rem  51
500 DATA "QUESTA E' UN'IMMAGINE MULTICOLO
    R"                           :rem  77
510 DATA "CREATA CON LE ROUTINE DI LINEA
    E FILL",                     :rem 201
520 !GR.7+16:!CO.1:N=32:FORI=0TO2*πSTEP π
    /N                           :rem  75
530 !CO.1:!PL.80,50:!DR.80+COS(I)*40,50-S
    IN(I)*32:NEXT                :rem 209
540 N=16:!CO.2:FORI=0TO2*πSTEPπ/N:X=80+
    COS(I)*50:Y=50-SIN(I)*40     :rem  73
550 !PL.X,Y:!DR.80+COS(I+π/N)*50,50-SIN(
    I+π/N)*40:NEXT               :rem 185
560 !CO.3:!PL.0,0:!DR.159,0:!DR.159,99:!D
    R.0,99:!DR.0,0               :rem 110
590 FORI=1TO1000:GETA$:IFA$=""THENNEXT
                                 :rem  60
620 !GR.7:!GR.0:RUN              :rem  56
630 REM                          :rem 125
640 FORI=1TO1000:GETA$:IFA$=""THENNEXT
                                 :rem  56
650 !GR.7+32+16:!SE.0,2,8:!SE.1,5,8:!SE.2
    ,0,14                        :rem  70
660 FORI=1TO1000:GETA$:IFA$=""THENNEXT
                                 :rem  58
670 GOTO750                      :rem 114
690 :                            :rem 217
700 PRINT"{CLR}{GIU'}":!GR.0:!SE.1,1,15:!
    SE.2,0,0:!SE.4,0,0           :rem 183
```



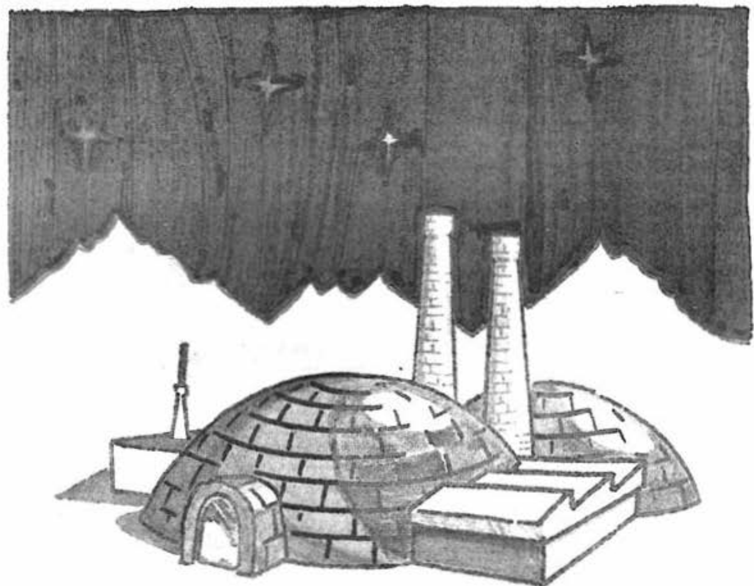
```
710 READN$:IFN$=""THEN730              :rem 171
720 PRINTCHR$(14)TAB(20-LEN(N$)/2)N$"
    {GIU'}":K=K+1:GOTO710              :rem 210
730 PRINTCHR$(14)TAB(11)"[< 16 @>]":PRINT
    TAB(11)"{RVS} PREMI UN TASTO {SU}"
                                       :rem 207
740 FORI=1TO1000:GETA$:IFA$=""THENNEXT
                                       :rem  57
750 !GR.8+16:!SE.2,0,0:!SE.1,RND(TI)*15,1
    0:!CO.1:RETURN                     :rem  44
```



La fabbrica gelata

di **C. Din**
trad. ed adatt.
di **L. Priotto**

È molto stressante lavorare da soli e fino a tardi in questo strano laboratorio di ricerca, dove la temperatura è sempre sotto zero. Quando decidete di tornarvene a casa vi accorgete però che dovete fare i conti con tre robot impazziti...



È circa mezzanotte alla Cybernetics Research and Development Plant, conosciuta anche come "la Fabbrica Gelata" a causa della sua costante temperatura di parecchi gradi sotto lo zero, richiesta dagli esperimenti di ricerca nel campo della robotica. Siete rimasti fino a questa tarda ora, ormai soli, per finire di lavorare a un progetto, ma a questo punto avete troppo freddo e decidete di rientrare a casa. Un ultimo controllo per assicurarsi che i generatori principali siano spenti, ma... che cosa si sta muovendo nel capannone?

Vi accorgete che quattro robot prototipi del tipo MK4 da combattimento sono ancora attivi per un difetto nei circuiti e vagano per il capannone alla ricerca di calore. Per vostra somma sfortuna l'unica fonte di calore del luogo siete voi stessi. Non potete distruggere i robot, mentre questi possono uccidere voi, semplicemente con un colpetto delle loro potentissime braccia di acciaio... Come fare? L'unico sistema per uscirne vivi è fare appello alla propria strategia.

Lo scopo del gioco è infatti sopravvivere il più a lungo possibile. Nel capannone sono disposti a caso dei grossi cubi di ghiaccio, che rappresentano la vostra unica difesa: potete infatti temporaneamente "congelare" un robot gettandogli contro uno di questi cubi di ghiaccio.

Per lanciare un cubo è sufficiente corrergli contro nella direzione in cui si desidera lanciarlo. Se due cubi di ghiaccio si trovano in posizione adiacente e spostate il primo contro il secondo, il primo cubo scomparirà e voi guadagnerete 10 punti. Guadagnate invece 1000 punti se riuscite a "congelare" un robot. Per ogni quattro robot "congelati" avanzate di livello, ottenete in bonus un numero di punti pari al tempo rimanente più il numero di cubi rimasti e guadagnate una vita extra, oltre ad un nuovo quantitativo di cubi di ghiaccio.

A livelli superiori al primo il contatore del tempo rimanente si decrementa tuttavia in modo più rapido. Infine vengono regalati 25 punti per ogni cubo lasciato nel capannone.

La fabbrica gelata

```
10 PRINT"{CLR}{ 12 GIU'}"SPC(14)"ATTENDER
   E..":POKE53272,28:POKE56,48:CLR
                                :rem 242
20 POKE56334,0:POKE1,51:A=2047:B=12288:C=
   53248                        :rem  33
30 FORI=.TOA:POKEI+B,PEEK(I+C):NEXT:POKE1
   ,55:POKE56334,1              :rem 206
40 EG=0:RESTORE:GOTO520         :rem 197
50 IF((PEEK(P1)AND2)=.)THEND=W:GOTO190
                                :rem  17
60 IF((PEEK(P1)AND4)=.)THEND=-T:G=2:GOTO1
   90                           :rem  46
70 IF((PEEK(P1)AND1)=.)THEND=-W:GOTO190
                                :rem  63
80 IF((PEEK(P1)AND8)=.)THEND=T:G=.:GOTO19
   0                            :rem   3
90 FORA=.TO3:IFZ>E(A)THENIFPEEK(E(A)+W)=N
   THENI=W:GOTO110              :rem  51
100 IFZ<E(A)THENIFPEEK(E(A)-W)=NTHENI=-W
                                :rem 224
110 IFZ>E(A)THENIFPEEK(E(A)+T)=NTHENI=T:G
    OTO130                      :rem 181
120 IFZ<E(A)THENIFPEEK(E(A)-T)=NTHENI=-T
                                :rem 220
130 IFPEEK(E(A)+I)=NTHENPOKEE(A),N:E(A)=E
    (A)+I:POKEE(A)+C,T:POKEE(A),5
                                :rem 134
140 IFE(A)+I=ZTHEN260           :rem 205
150 POKEE(A),4:NEXT             :rem 117
160 HH=HH+1:IFHH>HTTHENHH=.:H=H-1:IFH<0TH
    EN310                       :rem  82
170 PRINT"{HOME}{GIU'}{ 18 DES}";H;"{SIN}
    "                           :rem  54
180 GOTO50                      :rem  55
190 POKEU2,40:IFPEEK(Z+D)=NTHENPOKEZ,N:Z=
    Z+D:POKEZ+C,7:POKEZ,G:POKEU2,.:GOTO23
    0                           :rem  91
200 POKEU2,.:IFPEEK(Z+D)=6ANDPEEK(Z+D*2)<
    >NTHEN240                   :rem  24
210 IFPEEK(Z+D)=6ANDPEEK(Z+D*2)=NTHENQQ=Z
    +D:GOTO350                  :rem 175
220 IFPEEK(Z+D)<6THEN260        :rem 155
230 GOTO90                      :rem  55
240 FORA=.TO3:POKEZ+D,8:POKEU1,152:POKEZ,
    G:POKEZ+D,9:POKEU1,175:POKEZ,G+T:NEXT
                                :rem 251
250 POKEZ+D,N:POKEU1,.:S=S+10:PRINT"
    {HOME}{YEL}>{WHT}";S:GOTO90 :rem 100
260 FORY=1TO15:POKEZ,G+T:POKEU1,161:POKEE
    (A),4:POKEZ,G:FORO=1TO20:NEXT
                                :rem 233
270 POKEE(A),5                  :rem   0
280 NEXT:POKEU1,.:FORY=1TO15:POKEV,Y:POKE
    U1,245:FORO=1TO25:NEXT:NEXT:POKEU1,.
                                :rem  48
290 SH=SH-1:PRINT"{HOME}{ 2 GIU'}
    { 3 SIN}{OFF}";SH"{SIN} ";:IFSH=.THEN
    310                         :rem 153
300 GOTO50                      :rem  49
310 POKEZ,N                     :rem 151
320 PRINT"{HOME}{ 11 GIU'}{DES}
    { 14 SPAZI}{RVS}{WHT}GAME OVER{OFF}
    { 13 SPAZI}"                :rem  80
330 PRINT"{ 12 GIU'}{RVS}{ 12 DES}[<1>]PR
    EMI FUOCO...{ 2 SPAZI}";    :rem 224
340 WAIT56320,16,16:GOTO40      :rem 201
350 POKEU2,100:POKEU2,120:POKEQQ,N:QQ=QQ+
    D                           :rem 105
360 IFPEEK(QQ)=NTHENPOKEQQ+C,3:POKEQQ,6:G
    OTO350                      :rem  16
370 POKEU2,.:IFPEEK(QQ)<6THENPOKEQQ+C,3:P
    OKEQQ,6:S=S+1000:PRINT"{HOME}{YEL}>
    {WHT}";S:GOTO390            :rem 220
380 QQ=QQ-D:POKEQQ+C,3:POKEQQ,6:GOTO50
                                :rem 170
390 FORY=120TO135:POKEU2,Y:NEXT:POKEU2,.
                                :rem  32
400 FORA=.TO9:POKEQQ+C,T:POKEQQ,11:POKEU2
    ,135:POKEZ,G+T:POKEQQ,10:POKEZ,G
                                :rem 205
410 POKEU2,100:POKEU2,.:NEXT:POKEQQ,4:POK
    EU2,.:EG=EG+1:IFEG>3THEN430 :rem 247
420 GOTO50                      :rem  52
430 EG=.:FORY=64TO128:POKEU2,Y:NEXT:POKEZ
    ,N:FORA=.TO3:POKEE(A),N:NEXT:Z=1524
                                :rem 139
440 FORY=.TO3:FORO=35TO55:POKEU2,O:FORA=1
    TO10:NEXT:NEXT:NEXT:POKEU2,.:rem  47
450 FORY=1902TO1185STEP-1       :rem 238
460 IFPEEK(Y)=6THENPOKEY+C,T:POKEU2,120:F
    ORTD=1TO25:NEXT:POKEY+C,3:S=S+25
                                :rem 167
470 POKEU2,.:PRINT"{HOME}{YEL}>{WHT}";S:H
    =60:NEXT:FORY=5TO25:POKEU2,Y:NEXT:POK
    EU2,.:POKEZ,G               :rem  58
480 POKEZ+C,7:HT=HT-T:IFHT<1THENHT=1
                                :rem  49
490 SH=SH+T:PRINT"{HOME}{ 2 GIU'}
    { 3 SIN}";SH:LV=LV+1:PRINT"{HOME}
    {GIU'}{ 3 SIN}";LV          :rem 118
500 FORY=1TO10:B=1184+INT(RND(T)*700):IFP
    EEK(B)=NANDB<>ZTHENPOKEB+C,3:POKEB,6
                                :rem 238
510 NEXT:POKEU2,72:S=S+(H*10):PRINT"
    {HOME}{YEL}>{WHT}";S:POKEU2,.:GOTO50
                                :rem 117
520 POKE53281,1:POKE53272,21:PRINT"{CLR}
    { 5 GIU'}{DES}{CYN}{RVS}{ 9 DES}LA FA
    BBRICA GELATA!!{OFF}{BLK}"  :rem 176
530 PRINT"{ 3 GIU'}"SPC(17-LEN(STR$(S))/2
    )"PUNTI: "S:IFS>HSTHENHS=S  :rem 106
540 S=0:PRINT"{ 3 GIU'}"SPC(15-LEN(STR$(H
    S))/2)"HIGH SCORE: "HS      :rem 176
550 POKE53280,4:PRINT"{ 7 GIU'}{RVS}
    { 11 DES}PREMI [<1>]FUOCO{BLK} PER GI
    OCARE"                      :rem 233
560 FORA=54272TO54296:POKEA,0:NEXT:V=5429
    6:U1=54273                  :rem   9
570 U2=54280:POKE54277,17:POKE54284,17:PO
    KE54278,255:POKE54285,255   :rem 225
580 POKE54276,129:POKE54283,33  :rem 157
590 WAIT56320,16,0:WAIT56320,16,16
                                :rem  87
600 PRINT"{CLR}":POKE53281,0:FORX=12544TO
    12561:POKEX,.:NEXT          :rem 174
610 FORX=12288TO12383:READD:POKEX,D:NEXT:
    POKE53272,28                :rem   2
620 T=1:W=40:PRINT"{CLR}{YEL}>{WHT} 0":PO
    KEV,15                      :rem 140
630 PRINT"{ 8 SIN}{RVS}{GRN}LIV: {WHT}
    {OFF} 1":PRINT"{ 13 DES}{RVS}[<1>]TEM
    PO{OFF}{WHT} 60":C=54272    :rem 198
640 FORA=TTO70:POKEU2,20:B=1184+INT(RND(T
    )*700):POKEB+C,3:POKEB,6:POKEU2,.:NEX
    T                           :rem  91
```




```
650 FORA=1144TO1183:POKEA,7:POKEA+C,4:NEX
    T:FORA=1184TO1904STEPW:POKEA+C,4
                                :rem   33
660 POKEA,7:POKEA+39,7:POKEA+39+C,4:NEXT:
    FORA=1905TO1943:POKEA+C,4:POKEA,7:NEX
    T                           :rem  123
670 FORA=1TO15:POKEV,A:POKEU2,5:FORY=1TO7
    5:NEXT:NEXT:POKEU2,.        :rem  236
680 N=32:P1=56320:Z=1524:H=60:SH=3:E(.)=1
    510:E(T)=1534:E(2)=1724     :rem  202
690 E(3)=1204:POKEZ+C,7:POKEZ,.:HT=4:LV=1
                                :rem  174
700 PRINT"{HOME}{ 2 GIU'}{ 4 SIN}{YEL}@
    {WHT}";SH:GOTO50            :rem  162
710 PRINT"{HOME}{ 2 GIU'}{ 4 SIN}{YEL}@
    {WHT}";SH                   :rem  203
720 DATA 62,120,119,112,28,96,88,54,62,12
    0,127,60,112,120,48,28,124,30,238,14
                                :rem  199
730 DATA 56,6,26                :rem   21
740 DATA 108,124,30,254,60,14,30,12,56,60
    ,90,36,24,60,90,36,102,60,90,36,90
                                :rem   89
750 DATA 60,24,36,195           :rem   14
760 DATA 126,195,153,165,165,153,195,126,
    204,51,204,51,204,51,204,51,0,34,0
                                :rem   96
770 DATA 136,0,34,0             :rem  157
780 DATA 136,68,0,17,0,68,0,17,0,0,24,60,
    126,219,219,126,60,0,0,60,126,255
                                :rem   34
790 DATA 153,126,60             :rem  170
```





Alpha Anxiety

di **C. Howarth**
trad. ed adatt.
di **S. Colombo**

Ecco un gioco educativo per i più piccoli, che insegna a riconoscere le lettere dell'alfabeto. È richiesto l'uso di un joystick.

Viene visualizzata una griglia contenente tutte le lettere dell'alfabeto, tranne la Z, piazzate a caso nelle caselle della griglia stessa. Con il joystick il bambino sposta un cursore sopra la griglia, che man mano rivela le lettere nascoste. Quando il bambino trova la lettera da ricercare (quella che il computer visualizza in alto a sinistra) dovrà premere il pulsante di fuoco per selezionarla, visualizzandola in modo permanente.

Esiste un limite di tempo per trovare tutte le lettere dell'alfabeto: si inizia con 150 secondi, mentre tutti i round successivi avranno un limite di tempo basato su quello realizzato nel round precedente. Se si sbaglia nel selezionare una lettera, vengono detratti 10 secondi dal tempo rimanente, mentre ogni scelta corretta frutta 10 punti.

Al raggiungimento dei 1500 punti le lettere dovranno essere trovate in ordine inverso, mentre dopo i 2500 punti si dovranno trovare le lettere, sempre in ordine inverso, totalmente a memoria (senza cioè l'aiuto della lettera visualizzata in alto a sinistra).

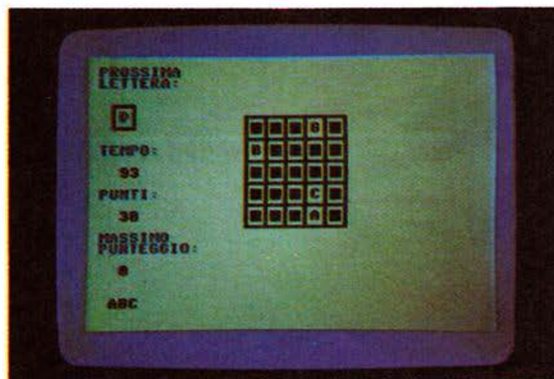
Ai 5000 punti viene calcolato il tempo medio realizzato per ogni round, tempo che viene assunto come limite per tutti i round seguenti, meno 5 secondi che vengono sottratti ad ogni 1000 punti ottenuti oltre i 5000.



Alpha Anxiety

```
100 SD=54272:FORI=SDTOSD+24:POKEI,0:NEXT:
    POKESD+5,26:POKESD+6,191:POKE54296,15
                                :rem  26
110 SM=1280:CL=54272:Q1=1905:Q2=1227
                                :rem 210
120 HS=0:JC=56320               :rem 159
130 CR$=CHR$(19):FORI=1TO23:CR$=CR$+CHR$(
    17):NEXT:POKE53281,1        :rem  54
140 FORI=1TO7                   :rem  13
150 PRINTCHR$(31)CHR$(147)LEFT$(CR$,9)SPC
    (12)"{ 4 GIU'}ALPHA ANXIETY":rem  47
160 FORJ=1TO50:NEXT             :rem 183
170 PRINTCHR$(31)CHR$(147)LEFT$(CR$,9)SPC
    (12)"{ 4 GIU'}{RVS}ALPHA ANXIETY"
                                :rem  67
180 FORJ=1TO100:NEXTI           :rem  46
190 DIMP%(25),RL%(5,5)          :rem 119
200 GOSUB1150                   :rem 217
```

```
210 RF=0:AT=0                          :rem  212
220 DL=150                             :rem  247
230 SC=0:MA=3                          :rem  208
240 L1=1:L2=1:LC=1:FL=1:PO=SM:FG=0
                                       :rem    2
250 IFRF=1THENLC=25                    :rem  142
260 PRINTCHR$(147)                     :rem   19
270 PRINT"{ 4 GIU'}"TAB(15)"{BLK}[<A>]*
    [<R>]*[<R>]*[<R>]*[<R>]*[<S>]"
                                       :rem  246
280 FORI=1TO4:PRINTTAB(15)"-{RVS} {OFF}-
    {RVS} {OFF}-{RVS} {OFF}-{RVS} {OFF}-
    {RVS} {OFF}-":PRINTTAB(15)"[<Q>]*+*+
    *+*+*[<W>]"                        :rem   45
290 NEXT:PRINTTAB(15)"-{RVS} {OFF}-{RVS}
    {OFF}-{RVS} {OFF}-{RVS} {OFF}-{RVS}
    {OFF}-"                            :rem  213
300 PRINTTAB(15) "[<Z>]*[<E>]*[<E>]*[<E>]
    *[<E>]*[<X>]"                      :rem  224
310 PRINT"{BLU}{HOME}{GIU'}{DES}PROSSIMA"
    :PRINT"{DES}LETTERA:":PRINT"{RED}
    {GIU'}{ 2 DES}[<A>]*[<S>]"         :rem   54
315 PRINT"{RED}{ 2 DES}- -":PRINT"{RED}
    { 2 DES}[<Z>]*[<X>]"               :rem    5
320 PRINT"{BLU}{GIU'}{DES}TEMPO:":PRINT"
    {BLU}{ 3 GIU'}{DES}PUNTI:":PRINT"
    {RED}{GIU'}{ 2 DES}"SC:            :rem  248
330 PRINT"{BLU}{GIU'}{DES}MASSIMO":PRINT"
    {DES}PUNTEGGIO:"                   :rem  225
340 PRINT"{RED}{GIU'}{ 2 DES}";HS
                                       :rem  165
350 IFSC<2499THENPOKEQ2+CL,6:POKEQ2,LC
                                       :rem  117
360 FORI=Q1+CLTOQ1+CL+26:POKEI,0:NEXT
                                       :rem   59
370 TI$="000000":CS=160                :rem  160
380 GOSUB530                           :rem  179
390 IFFG=1THEN240                      :rem  235
400 IFMA=0THEN450                      :rem  230
410 POKESD+4,17:POKESD+1,0             :rem    6
420 TL=DL-INT(TI/60)                   :rem  158
430 PRINTLEFT$(CR$,11)SPC(2)TL"{SIN}
    { 2 SPAZI}"                        :rem  245
440 IFTL>0THEN380                      :rem  255
450 IFSC>HSTHENHS=SC                   :rem   52
460 PRINT"{CLR}{BLU}{ 9 GIU'}{ 15 SPAZI}
    {RVS}GAME OVER"                    :rem   30
470 PRINT"{BLU}{ 3 GIU'}{ 9 SPAZI}PREMI F
    UOCO"                              :rem  183
480 PRINT"{GIU'}{ 8 SPAZI}PER GIOCARE ANC
    ORA"                               :rem   19
490 POKESD+1,85:FORI=1TO400:NEXT:POKESD+1
    ,0:GOSUB1150                       :rem   25
500 A=NOTPEEK(JC)AND16:IFA=0THEN500
                                       :rem  124
510 PRINTCHR$(147):FORT=1TO500:NEXT
                                       :rem  165
520 GOTO210                            :rem   99
530 P=NOTPEEK(JC)AND31                 :rem   80
540 JE=(P=8)                           :rem  123
550 JS=(P=2)                           :rem  132
560 JW=(P=4)                           :rem  139
570 JN=(P=1)                           :rem  128
580 JF=(P=16)                          :rem  175
590 TH=PO:T1=L1:T2=L2                  :rem   10
600 IFJFTHENGOSUB740:GOTO670           :rem   15
610 IFJETHENPO=PO+2:L1=L1+1:FL=1:rem  195
620 IFJSTHENPO=PO+80:L2=L2+1:FL=1
                                       :rem   10
630 IFJWTHENPO=PO-2:L1=L1-1:FL=1:rem  219
640 IFJNTHENPO=PO-80:L2=L2-1:FL=1
                                       :rem   11
650 IFFL=0THEN730                      :rem  242
660 POKESD+1,50                        :rem   79
670 IFFG=1THEN730                      :rem  240
680 IFPO<SMORPO>SM+376THENPO=TH:L1=T1:L2=
    T2                                 :rem   44
690 IFPEEK(PO+1)=32ORPEEK(PO-1)=32THENPO=
    TH:L1=T1:L2=T2                     :rem  146
700 POKETH+CL,0:POKETH,CS:CS=PEEK(PO)
                                       :rem   97
710 POKEPO+CL,2:POKEPO,RL%(L1,L2)
                                       :rem  237
720 FL=0                               :rem  152
730 RETURN                             :rem  122
740 IFRL%(L1,L2)=LCTHENSC=SC+10:GOTO780
                                       :rem  103
750 IFRF=1THEN860                      :rem  254
760 IFRL%(L1,L2)<LCTHEN900             :rem  250
770 GOTO860                            :rem  117
780 POKESD+1,100:FORT=1TO25:NEXT:POKESD+1
    ,0                                 :rem  156
790 IFRF=1THENPOKEQ1+26-LC,LC:GOTO810
                                       :rem   43
800 POKEQ1+LC,LC                       :rem  190
810 IFRF=1THENLC=LC-1:GOTO830          :rem   36
820 LC=LC+1                            :rem   81
830 IFSC>2500THEN850                   :rem  145
840 IFLC<>0THENPOKEQ2,LC               :rem    0
850 CS=RL%(L1,L2):GOTO890              :rem  191
860 POKESD+1,20:FORT=1TO30:NEXT:POKESD+1,
    0                                  :rem  104
870 DL=DL-10:IFDL<=0THENGOTO460        :rem  142
880 RETURN                             :rem  128
890 PRINTLEFT$(CR$,15)SPC(2)SC"{SIN}
    { 2 SPAZI}"                        :rem  249
900 IFRF=0THEN930                      :rem  248
910 IFLC<>0THEN1140                    :rem   87
920 GOTO940                            :rem  113
930 IFLC<>26THEN1140                   :rem  145
940 IFSC>5000THEN1000                  :rem  181
950 AT=AT+DL-TL                        :rem  141
960 IFSC=5000THENDL=INT(AT/20):GOTO1050
                                       :rem   93
```



```
970 IFTL<150THENDL=40+TL*2:GOTO1050
                                    :rem  97
980 IFTL>300THENDL=TL*.5:GOTO1050
                                    :rem   3
990 DL=TL                           :rem  15
1000 IFSC=6000THENDL=DL-5           :rem 215
1010 IFSC=7000THENDL=DL-5           :rem 217
1020 IFSC=8000THENDL=DL-5           :rem 219
1030 IFSC=9000THENDL=DL-5           :rem 221
1040 IFSC=10000THENDL=DL-5          :rem   6
1050 H=SD+1:L=SD:POKEH,8:POKEL,97:GOSUB10
     80                             :rem 129
1060 POKEH,12:POKEL,143:GOSUB1080:POKEH,1
     4:POKEL,24:GOSUB1080           :rem 213
1070 POKEH,12:POKEL,143:POKESD+4,17:FORI=
     1TO800:NEXTI:POKEH,0:POKEL,0:GOTO111
     0                              :rem  99
1080 POKESD+4,17:FORI=1TO200:NEXT:POKESD+
     4,16:FORI=1TO100:NEXT          :rem 127
1090 POKESD+4,17:FORI=1TO200:NEXT:POKESD+
     4,16:FORI=1TO100:NEXT          :rem 128
1100 RETURN                         :rem 162
1110 IFSC>1250THENRF=1              :rem  35
1120 GOSUB1150                      :rem  11
1130 FG=1                           :rem 192
1140 RETURN                         :rem 166
1150 FORI=1TO25:P%(I)=I:NEXT        :rem 183
1160 FORI=1TO5                      :rem  62
1170 FORJ=1TO5                      :rem  64
1180 R=INT(RND(1)*25+1)             :rem 232
1190 IFP%(R)=0THEN1180              :rem 216
1200 RL%(I,J)=R                     :rem  37
1210 P%(R)=0                        :rem  73
1220 NEXTJ                          :rem  78
1230 NEXTI                          :rem  78
1240 RETURN                         :rem 167
```

SUL PROSSIMO NUMERO TROVERETE:

– Italian Wars: uno splendido wargame con ottima grafica, in cui dovrete conquistare l'Italia per renderne possibile l'unità.
– 120 colonne per MPS 801-803: un programma in linguaggio macchina per rendere professionali le prestazioni della vostra stampante.
– Il drive SDF-1001: tutto ciò che occorre sapere sul nuovo drive Commodore da 1 Megabyte.
– Due piacevoli sorprese natalizie: una allegata alla rivista e l'altra su una facciata della cassetta...
E, come sempre, altri programmi di utilità e di svago per il vostro C64 assetato di SUPER!

Riduttore e velocizzatore di programmi

di **J. Wang**
trad. ed adatt.
di **L. Priotto**

Vi sarà certamente capitato di possedere programmi BASIC che occupano quasi tutta la memoria disponibile del computer e, proprio per questo, sono affetti da una notevole lentezza di esecuzione. Con il programma qui presentato sarete in grado di aumentare la quantità di memoria libera e, nel contempo, aumentare sensibilmente la velocità di esecuzione del programma.



Nella programmazione in BASIC vigono due regole importanti: la prima dice che maggiore è la quantità di memoria occupata dal programma, minore sarà la sua velocità di esecuzione. La seconda – del resto collegata indirettamente alla prima – dice che un programma è tanto più veloce quanto più le sue linee sono "fitte" di istruzioni. Per esemplificare, dato il programma:

```
10 PRINTCHR$(147)
20 DIMB(100)
30 FORI=1TO100
40 A=INT(RND(0)*100)
50 B(I)=A
60 PRINTB(I),
70 NEXTI
```

esso avrà un tempo di esecuzione impercettibilmente migliore, se scritto nella forma:

```
10 PRINTCHR$(147):DIMB(100):
   FORI=1TO100:A=INT(RND(0)*
   100):B(I)=A:PRINTB(I),:NE
   XT
```

Il guadagno in tempo di esecuzione è tanto maggiore quanto più è lungo il programma stesso.

Il programma "Riduttore", presentato in fondo a questo articolo, esegue esattamente questa operazione: trasforma un programma con poche istruzioni per linea in un programma estremamente compatto, e quindi più veloce, oltre che più "corto".

Il programma "Espansore" esegue invece il procedimento inverso: da un programma con molte istruzioni per linea ne ricava una versione con poche istruzioni per linea.

Entrambi i programmi richiedono l'utilizzo del disk drive, ed operano leggendo le linee di un dato programma su disco, elaborandole in modo da compattarle (o espanderle) il più possibile, per riscriverle quindi sul disco sotto il nome di un altro file.

Se ad esempio desiderate compattare il programma su disco dal nome "PIPPO", il programma Riduttore ne esaminerà le linee (visualizzandone i numeri sullo schermo) e scriverà su disco un altro file "compattato" dal nome "PIPPO/C". Se si desidera espandere il programma "PIPPO", si userà il programma Espansore, che, con funzionamento analogo al precedente, creerà un altro file su disco dal nome "PIPPO/U".

Come vedete, il file originario non viene né riscritto né cancellato, ma viene semplicemente scritto sul disco un altro file con le caratteristiche desiderate. Naturalmente, il procedimento di compattazione e di espansione non è una cosa facilissima, quindi il computer impiegherà un tempo piuttosto lungo a concludere l'operazione. Un po' di pazienza verrà comunque compensata dal risultato ottenuto.

Un'ultima avvertenza: come quasi tutti sanno, l'editazione di una linea BASIC è possibile solo se questa non supera gli 80 caratteri, mentre il programma Riduttore riesce a compattare una linea fino a 255 caratteri. Non sarà quindi possibile modificare in alcun modo le linee che superano il limite di 80 carat-

teri, pena la perdita definitiva della parte di linea eccedente questo valore.
Se si desidera modificare alcune linee di programma compattate ad un numero di caratteri superiore al massimo "editabile", sarà necessario espandere il programma compattato con l'Espansore, modificare le linee opportune e quindi ricompattarlo nuovamente.

Riduttore

```
10 DIMTL(1000):INPUT"{CLR}NOME DEL FILE S
   U DISCO";FL$:PRINT"{GIU'}SCANSIONE IN
   CORSO:"                         :rem 211
20 OPEN15,8,15:GOSUB730:OPEN5,8,5,"0:"+FL
   $+",P,R":GOSUB730:GOSUB700      :rem 115
30 GOSUB700:IFV+V1=0THEN190        :rem 129
40 GOSUB700:LN=V1+(256*V)          :rem 172
50 GOSUB710                        :rem 125
60 IFV=0THEN30                     :rem  74
70 IFV=137ORV=141THEN90            :rem 134
80 IFV<>167THEN50                  :rem 249
90 LT=0                            :rem 118
100 GOSUB710:IFV=32THEN100         :rem  42
110 IFV<48ORV>57THEN130            :rem 131
120 LT=(10*LT)+VAL(C$):GOSUB710:GOTO110
                                   :rem   9
130 FORX=0TON:IFTL(X)=LTTHEN160    :rem 230
140 NEXT:TL(N)=LT:N=N+1:PRINTLT,:IFN<1000
    THEN160                        :rem   6
150 PRINT:PRINT"{RVS}TROPPE LINEE OGGETTO
    !":GOTO750                     :rem 208
160 IFV=44THEN90                   :rem 185
170 IFV<>32THEN60                  :rem 241
180 GOSUB710:GOTO160               :rem 187
190 IFN<2THEN230                   :rem 169
200 FORX=0TON-1:FORY=0TON-2:IFTL(Y)<TL(X)
    THEN220                        :rem  31
210 V=TL(Y):TL(Y)=TL(X):TL(X)=V    :rem 144
220 NEXT:NEXT                      :rem  76
230 PRINT:PRINT"{GIU'}RIDUZIONE IN CORSO:
    ":CLOSE5:OPEN5,8,5,"0:"+FL$+",P,R":GO
    SUB730                         :rem  17
240 FO$=LEFT$(FL$,14)+"/C":PRINT#15,"0:"+
    FO$:OPEN6,8,6,"@0:"+FO$+",P,W"
                                   :rem  54
250 GOSUB730:GOSUB700:PRINT#6,CHR$(V1);
                                   :rem  98
260 PRINT#6,CHR$(V);:R=0           :rem 134
270 GOSUB700:K1=V1:K2=V:F=0:IFV+V1=0THEN6
    90                             :rem 109
280 GOSUB700:L1=V1:L2=V:LN=L1+(256*L2):PR
    INTLN,                         :rem  83
290 GOSUB710:IFV=32ORV=58THEN290:rem 223
300 IFV=0THEN330                   :rem 170
310 IFV<>143THEN360                :rem  83
320 GOSUB710:IFV>0THEN320          :rem 254
330 F=1:FORX=0TON:IFTL(X)<LNTHENNEXT
                                   :rem 119
340 IFTL(X)=LNTHEN360              :rem  14
350 GOTO270                        :rem 106
360 PRINT#6,CHR$(K1);CHR$(K2);:PRINT#6,CH
    R$(L1);CHR$(L2);:R=4:IFFTHENPRINT#6,"
    :";:R=5                        :rem  35
370 F=0:GOTO400                    :rem  84
380 PRINT#6,C$;:R=R+1              :rem 198
390 GOSUB710                       :rem 180

400 IFV=137THENF=1                 :rem  52
410 IFV=139ORV=167THENF=1          :rem   9
420 IFV=0THEN570                   :rem 179
430 IFV=32THEN390                  :rem 233
440 IFV<>143THEN470                :rem  89
450 GOSUB710:IFV>0THEN450          :rem   6
460 GOTO570                        :rem 111
470 IFV<>34THEN520                 :rem  39
480 PRINT#6,C$;:R=R+1:GOSUB710:IFV=34THEN
    380                            :rem 166
490 IFV>0THEN480                   :rem 187
500 IFFTHENV=0:GOTO260             :rem 103
510 PRINT#6,CHR$(34);:R=R+1:GOTO570

                                   :rem  34
520 IFV<>58THEN380                 :rem  45
530 GOSUB710:IFV=32ORV=58THEN530:rem 217
540 IFV=143THEN450                 :rem  27
550 IFV=0THEN570                   :rem 183
560 PRINT#6,":";:R=R+1:GOTO400     :rem 228
570 IFFOR(R>170)THENV=0:GOTO260    :rem 136
580 GOSUB700:IFV+V1=0THEN690       :rem 192
590 GOSUB700:LN=V1+(256*V):L1=V1:L2=V:PRI
    NTLN,:FORX=0TON:IFTL(X)<LNTHENNEXT

                                   :rem 102
600 IFTL(X)=LNTHEN650              :rem  15
610 GOSUB710:IFV=143THEN450        :rem 107
620 IFV=32ORV=58THEN610            :rem 134
630 IFV=0THEN580                   :rem 183
640 PRINT#6,":";:R=R+1:GOTO400     :rem 227

650 PRINT#6,CHR$(0);CHR$(1);CHR$(1);:PRIN
    T#6,CHR$(L1);CHR$(L2);:R=4     :rem  68
660 GOSUB710:IFV=32ORV=58THEN660:rem 225
670 IFV=0ORV=143THENPRINT#6,":";:rem 181
680 F=0:GOTO400                    :rem  88

690 PRINT#6,CHR$(0);CHR$(0);CHR$(0);:PRIN
    T"{CLR}{RVS}HO TERMINATO - OK.{OFF}":
    GOTO750                        :rem 120
700 GOSUB710:V1=V                  :rem   3

710 GET#5,C$:GOSUB730:IFC$=""THENV=0:RETU
    RN                             :rem 116
720 V=ASC(C$):RETURN               :rem 213

730 INPUT#15,EN,EM$,ET,ES:IFEN=0THENRETUR
    N                              :rem 181
740 PRINT:PRINT"{RVS}DISK ERROR{OFF}":PRI
    NTEN;EM$;ET;ES                 :rem 126
750 CLOSE5:CLOSE6:CLOSE15          :rem  67
```

Espansore

```
10 GOTO150                         :rem  48
20 GOSUB40                         :rem  70
30 V1=V                            :rem 125
40 GET#5,C$                        :rem  47
50 GOSUB90                         :rem  78
60 IFC$=""THENV=0:RETURN           :rem 233
70 V=ASC(C$)                       :rem 137
80 RETURN                          :rem  72
90 INPUT#15,EN,EM$,ET,ES           :rem 172
100 IFEN=0THENRETURN               :rem  47
110 PRINT"{CLR}{RVS}ERRORE SUL DISCO{OFF}
    "                              :rem 207
```

```
120 PRINT                              :rem  32
130 PRINTEN;EM$;ET;ES                  :rem  76
140 GOTO770                            :rem 108
150 PRINT"{CLR}"                       :rem 250
160 INPUT"{CLR}NOME DEL FILE SU DISCO";FL
    $                                  :rem  45
170 DIMC(256)                          :rem 163
180 OPEN15,8,15                        :rem  39
190 OPEN5,8,5,"0:"+FL$+",P,R"          :rem 234
200 GOSUB90                            :rem 123
210 PRINT                              :rem  32
220 PRINT"OK.STO LAVORANDO SULLA LINEA:
    {GIU'}"                            :rem 254
230 FO$=LEFT$(FL$,14)+"/U"             :rem 101
240 PRINT#15,"@0:"+FO$                 :rem 170
250 OPEN6,8,6,"0:"+FO$+",P,W"          :rem 241
260 GOSUB90                            :rem 129
270 GOSUB20                            :rem 123
280 PRINT#6,CHR$(V1);C$;               :rem  98
290 F=1                                :rem  79
300 GOTO400                            :rem  96
310 LN=NL                              :rem   5
320 IFLK=0THEN750                      :rem 243
330 PRINTLN,                           :rem 233
340 PRINT#6,CHR$(1);CHR$(1);           :rem  37
350 PRINT#6,CHR$(LL);CHR$(LH);         :rem 240
360 X=1                                :rem  95
370 GOSUB40                            :rem 126
380 C(X)=V                             :rem  26
390 IFV>0THENX=X+1:GOTO370             :rem 116
400 GOSUB20                            :rem 118
410 LK=V+V1                            :rem 113
420 IFLK=0THEN470                      :rem 243
430 GOSUB20                            :rem 121
440 NL=V1+(256*V)                      :rem 143
450 LL=V1                              :rem 245
460 LH=V                               :rem 193
470 IFFTHENF=0:GOTO310                 :rem  89
480 X=1                                :rem  98
490 IFC(X)<>58THEN600                  :rem 196
500 IFX=1THEN720                       :rem 178
510 LN=LN+1                            :rem  99
520 IFLN>=NLTHEN720                    :rem 157
530 PRINT#6,CHR$(0);CHR$(1);CHR$(1);
                                       :rem 227
540 H=INT(LN/256)                      :rem 192
550 L=LN-(256*H)                       :rem  74
560 PRINT#6,CHR$(L);CHR$(H);           :rem  91
570 X=X+1                              :rem 229
580 IFC(X)=32ORC(X)=58THEN570          :rem 188
590 GOTO490                            :rem 116
600 IFC(X)<128ORC(X)>155THEN670        :rem  26
610 IFC(X)=128ORC(X)>153THEN640        :rem  23
620 IFC(X)<137ORC(X)>144THEN670        :rem  26
630 IFC(X)=140ORC(X)=141THEN670        :rem  18
640 PRINT#6,CHR$(C(X));                :rem  37
650 IFC(X)>0THENX=X+1:GOTO640          :rem   9
660 GOTO310                            :rem 105
670 IFC(X)<>34THEN720                  :rem 193
680 PRINT#6,CHR$(C(X));                :rem  41
690 X=X+1                              :rem 232
700 IFC(X)=34ORC(X)=0THEN720           :rem 120
710 GOTO680                            :rem 111
720 PRINT#6,CHR$(C(X));                :rem  36
730 IFC(X)>0THENX=X+1:GOTO490          :rem  11
740 GOTO310                            :rem 104
750 PRINT#6,CHR$(0);CHR$(0);           :rem  40
760 PRINT"{CLR}HO TERMINATO - OK"
                                       :rem  18
770 CLOSE5:CLOSE6:CLOSE15              :rem  69
```

Dynamic SID editor

di **W. Eastwood**
trad. ed adatt.
di **L. Priotto**

Per usare il chip sonoro SID è necessario far ricorso ad un gran numero di PEEK e POKE. Questa utility vi fornisce un menu tale da poter scegliere istantaneamente i valori del SID e compiere quindi facilmente esperimenti di programmazione sonora.



Dopo aver attivato il programma appare un menu a pieno schermo, con tutti i registri del SID posti a zero e con l'editor impostato sulla voce 1. Per modificare un parametro della voce 1 è sufficiente premere il tasto corrispondente alla lettera evidenziata del parametro (esempio F per frequenza, A per attack, ecc.). Per cambiare voce è sufficiente premere il tasto di funzione corrispondente alla voce desiderata.
Esistono tre modi in cui possono essere modificati i vari parametri del SID.
Modo 1: alcuni parametri possono essere soltanto in due stati: on ed off. Ad esempio, la porta (Gate) di una voce può essere solo attiva (ed in questo caso è visualizzata in campo inverso) o disattiva.
Modo 2: altri parametri possono assumere un numero limitato di valori: se si preme ad esempio il tasto W per selezionare la forma d'onda, essa ciclerà attraverso i suoi possibili valori, visualizzando il valore correntemente selezionato.
Modo 3: infine altri parametri possono assumere valori da 0 a un valore massimo. Ad esempio, uno di questi è la Frequenza: premendo F, il parametro sarà posto in campo inverso, per evidenziarne la selezione; a questo punto quattro diversi tasti controllano il valore del parametro:
1. SHIFT: se premuto, fa in modo che l'incremento di valore sia positivo. Se non è premuto, l'incremento viene assunto come negativo.
2. CURSORE SU/GIÙ: provocano un incremento di valore pari a 1.
3. CURSORE DES/SIN: provocano un incremento di 29.
4. RETURN: provoca un incremento direttamente proporzionale al valore del parametro, con effetto logaritmico.
Si ricordi che per ottenere un suono (diverso dal cicalino che segnala un errore nella pressione dei tasti) il volume deve avere un valore diverso da zero, la voce deve avere una forma d'onda e un valore diverso da zero nell'attack, decay e/o sustain, e la porta (Gate) deve essere attiva. Se viene selezionata la forma d'onda pulsante, la larghezza d'impulso (Pulse width) deve avere un valore diverso da zero. Perché il suono sia udibile, ci deve inoltre essere una certa congruenza tra i vari parametri selezionati, congruenza che si trova con un po' di sperimentazione. A titolo di esempio si provi con i valori visualizzati nella fotografia.

ABBONATI A SUPE
NESSUNO

Vuoi ricevere a casa ogni mese le ultime novità software che riguardano il tuo amato computer? Abbonarsi a SUPERCOMMODORE C64 è il modo più sicuro e conveniente: puoi risparmiare il 20% rispetto a quello che spenderesti in edicola e puoi scegliere tra due formule di abbonamento. La formula "solo rivista" oppure la formula "rivista più cassetta", che ti evita di digitare i lunghi listati che servono per far "girare" i programmi pubblicati.

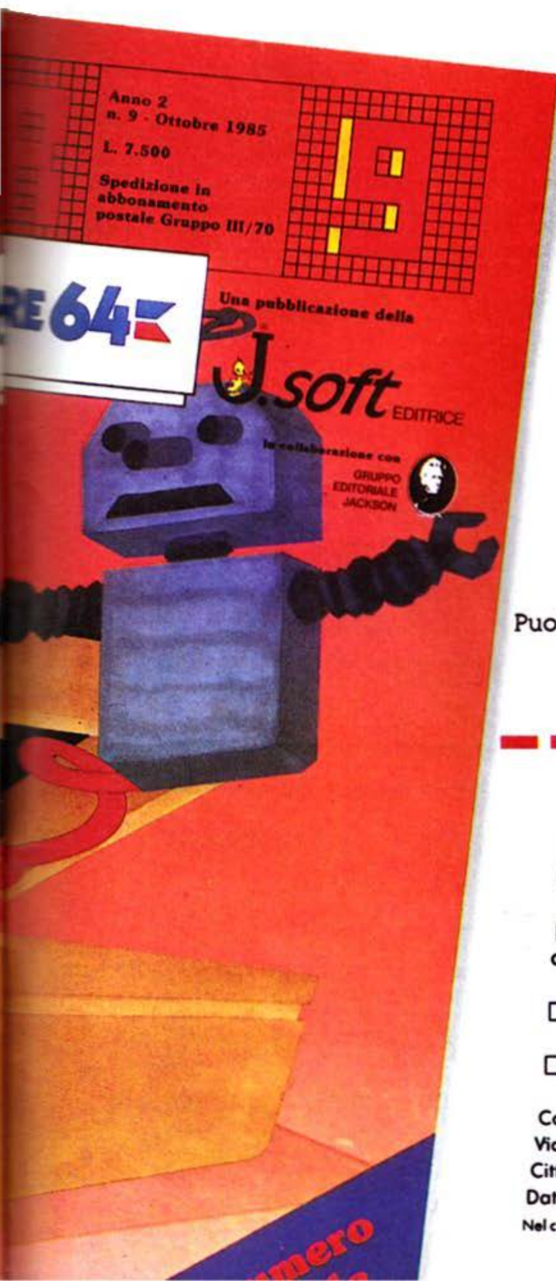
Abbonarsi a SUPERCOMMODORE C64 è sicuro e conveniente.

Il tuo computer ti ringrazierà.



J.soft s.r.l.
Viale Restelli, 5 - 20124 Milano
Tel. 02/683797-6880841-6880842-6880843-6888228

COMMODORE C64 LA SUPERA



Scegli la formula di abbonamento che preferisci e spedisci subito il tagliando di abbonamento alla tua Super rivista.

È il meno che tu possa fare per il tuo super computer.

Puoi effettuare il versamento sul C.C.P. n° 19445204 allegato intestato a J.*soft* - Milano, oppure ritaglia il tagliando sotto riportato e spediscilo in busta chiusa, allegando assegno intestato a J.*soft* - Milano.

Tagliando abbonamento a SUPERCOMMODORE C64

☐ Abbonamento a 11 numeri di SUPERCOMMODORE C64 (SC) L. 30.000 anziché L. ~~38.500~~

☐ Abbonamento a 11 numeri di SUPERCOMMODORE C64 con cassetta (SCC) L. 66.000 anziché L. ~~82.500~~

☐ Assegno allegato n° ____________

☐ Ho versato l'importo sul C.C.P. n° 19445204 allegato

Cognome ____________ Nome ____________
Via ____________
Città ____________ C.A.P. ____________ Prov. ______
Data ____________ Firma ____________

Nel caso di versamento su C.C.P. preghiamo di allegare fotocopia della ricevuta di versamento.

```
100 GOTO1000:REM
    ***** CALCOLI A 16 BIT   *****
                                  :rem  203
200 KK=PEEK(KEY):IFKK=NULLTHEN200
                                  :rem   92
205 REM TASTO PREMUTO; USA LE VARIABILI P
    ER AVERE MAGGIORE VELOCITA' :rem  157
210 J=BB:Z=PEEK(H+BB):HH=PEEK(SHFT)
                                  :rem  142
215 REM CURS SU=7,CURS SIN=2,RETURN=1
                                  :rem   17
220 ONKKGOTO240,250,230,230,230,230,260
                                  :rem   33
230 RETURN:REM TASTO NON VALIDO, MANTIENE
     IL VALORE                    :rem  163
240 J=INT((Z+BB)*K/CC):GOTO260   :rem  232
245 REM MOLTO VELOCE, PIU' ELEVATO PER VA
    LORI ALTI                     :rem  247
250 J=K:REM INCREMENTO MODERATO :rem  214
260 Z=PEEK(H)+Z*XX+J*((HH=AA)-(HH>AA))
                                  :rem  170
265 REM H SETTATO DALL'UTENTE; AGGIUNGE I
    NCREMENTO                     :rem  124
266 REM INCREMENTO POSITIVO SE TASTO SHIF
    T PREMUTO                     :rem  192
270 IFZ<AATHENZ=AA:REM VALORI MIN & MAX
                                  :rem   99
280 IFZ>ZZTHENZ=ZZ                :rem  239
290 GOSUB930:REM POSIZIONA IL CURSORE
                                  :rem   73
295 REM STAMPA VALORE SE FILTRO :rem  123
300 IFDV<>CCTHENPRINTZ"{SIN} ":GOTO320
                                  :rem  187
305 REM STAMPA % DEL DUTY CYCLE :rem  242
310 PRINTINT(Z/4.095)/10"{SIN}%
    { 2 SPAZI}"                   :rem  124
315 REM INSERISCE VALORE NEL BUFFER
                                  :rem  132
320 POKEH+BB,Z/XX:POKEH,Z-INT(Z/XX)*XX
                                  :rem  211
330 IFDV>CCTHEN360:REM SALTA PER FILTRO
                                  :rem  149
335 REM INSERISCE VALORE NEL SID E RITORN
    A PER NUOVI DATI              :rem  170
340 POKEYY,PEEK(H):POKEYY+BB,PEEK(H+BB)
                                  :rem  197
350 GOTO200                       :rem   99
355 REM FILTRO DISPARI; VALORE A 11 BIT
                                  :rem  227
356 REM IMMAGAZZINATI 3 BASSI, 8 ALTI
                                  :rem  138
360 POKEYY,ZAND7:POKEYY+BB,Z/8   :rem  193
370 GOTO200:REM
    ***** CALCOLI AD 8 BIT   *****
                                  :rem  186
400 KK=PEEK(KEY):IFKK=NULLTHEN400
                                  :rem   96
405 REM TASTO PREMUTO; E' VALIDO?
                                  :rem  217
410 ONKKGOTO430,430,420,420,420,420,430
                                  :rem   38
415 REM NO, RITORNA MANTENENDO IL VALORE
                                  :rem  173
420 RETURN                        :rem  118
430 HH=PEEK(SHFT):BB=SBUF+H      :rem  173
435 REM AGGIUNGE INCREMENTO (UGUALE PER T
    UTTI                          :rem   39
436 REM I TASTI VALIDI) POSITIVO SE SHIFT
     E' PREMUTO                   :rem   92
440 Z=(PEEK(BB)ANDJ)/K+(HH=0)-(HH>0)
                                  :rem    6
450 IFZ<0THENZ=0:REM VALORI MIN & MAX
                                  :rem  191
460 IFZ>15THENZ=15                :rem   83
470 GOSUB930:PRINTZ"{SIN}{ 2 SPAZI}":REM
    POSIZIONA E STAMPA            :rem  158
475 REM ATTIVA MASCHERA SU UN BYTE
                                  :rem   16
480 Z=PEEK(BB)AND255-JORZ*K      :rem  131
485 REM INSERISCE E RITORNA PER NUOVI DAT
    I                             :rem   40
490 POKEBB,Z:POKESID+H,Z:GOTO400:REM
    ***** DECODIFICA BIT *****   :rem  123
700 YY=PEEK(SBUF+XX):REM VALORE DEL BYTE
                                  :rem  247
710 Z=INT((YYANDJ)/K):REM VALORE DEI BIT
                                  :rem  157
715 REM VALORI DI MASCHEERA DEFINITI DALL
    ' UTENTE                      :rem  201
720 RETURN:REM
    ***** CODIFICA BIT *****     :rem   76
800 YY=YYAND255-JORZ*K:REM CODIFICA
                                  :rem  223
810 POKESBUF+XX,YY:REM SETTA IL BUFFER
                                  :rem  159
820 POKESID+XX,YY:REM SETTA IL SID
                                  :rem  118
830 RETURN:REM
    ***** PLOT DEL CURSORE *****:rem  116
900 POKECX,WX%(CT)+CL:REM RIGA DI UNA PAR
    OLA                           :rem  118
910 POKECY,WY%(CT)+OFS:REM E SUA COLONNA
                                  :rem  183
920 GOTO950                       :rem  114
930 POKECX,VX%(CT)+CL:REM VALORE RIGA
                                  :rem   17
940 POKECY,VY%(CT):REM E VALORE COLONNA
                                  :rem  134
950 SYSMOVE:RETURN:REM MUOVE IL CURSORE
                                  :rem   80
996 REM
    ** ROUTINE PRINCIPALE **     :rem   65
1000 GOSUB50000:REM INIZIALIZZA          *
     **** INPUT DA TASTIERA *****
                                  :rem  184
1100 KK=PEEK(KEY):IFKK=NULLTHEN1100
                                  :rem  188
1105 REM TASTO PREMUTO; SE SHIFT E' PREMU
     TO                           :rem   75
 106 REM SALTA ALLA SUCCESSIVA VERIFICA
                                  :rem  125
1110 HH=PEEK(SHFT):IFHH=1THEN1500
                                  :rem  247
1115 REM INSERZIONE LIVELLI NELLE VOCI?
                                  :rem  163
1120 IFKK<F1ORKK>F5THEN1500      :rem  129
1125 REM MODIFICA LIVELLO E PULISCE DISPL
     AY                           :rem  112
1130 EL=KK-4:DV=-1:GOSUB53000:GOTO1100
                                  :rem  224
1496 REM
     ***** ORGANIZZAZIONE TASTI *****
                                  :rem   27
1500 IFKK=FTHEN2000:REM FREQUENZA
                                  :rem   46
```

```
1510 IFKK=PTHEN2000:REM LARGHEZZA IMPULSO
                               :rem  89
1520 IFKK=ATHEN3000:REM ATTACK  :rem  51
1530 IFKK=DTHEN3000:REM DECAY   :rem 229
1540 IFKK=STHEN3000:REM SUSTAIN :rem 182
1550 IFKK=RTHEN3000:REM RELEASE :rem 144
1560 IFKK=WTHEN4000:REM FORMA D'ONDA
                               :rem 152
1570 IFKK=GTHEN4000:REM PORTA   :rem  13
1580 IFKK=ITHEN4000:REM RING    :rem 186
1590 IFKK=YTHEN4000:REM SYNC    :rem 216
1600 IFKK=QTHEN2000:REM FREQ FILTRO
                               :rem 135
1610 IFKK=NTHEN3000:REM RISONANZA
                               :rem  61
1620 IFKK=VTHEN3000:REM VOLUME  :rem 105
1630 IFKK=MTHEN5000:REM MODO    :rem 186
1640 IFKK=N1THEN5000:REM ASSEGNAZIONE
                               :rem  69
1650 IFKK=N2THEN5000            :rem 162
1660 IFKK=N3THEN5000            :rem 164
1670 IFKK=XTHEN5000             :rem 124
1680 IFKK=OTHEN6000:REM OSC3    :rem 171
1690 IFKK=ETHEN6000:REM ENV3    :rem 166
1700 IFKK=TTHEN5000:REM 3 OFF   :rem 158
1705 REM FUNZIONI SPECIALI      :rem 109
1710 IFKK=F1THENGOSUB7000:GOTO1100
                               :rem  77
1720 IFKK=F3THENGOSUB7000:GOTO1100
                               :rem  80
1730 IFKK=F5THENGOSUB7000:GOTO1100
                               :rem  83
1740 IFKK<>F7ORHH=0THEN1800:REM FINE?
                               :rem 253
1750 GETA$:IFA$<>""THEN1750:REM PULISCE I

     L BUFFER                  :rem 124
1760 POKESID+24,0:SYSCS:END:REM FINE
                               :rem 234
1800 DV=-1:GOSUB53020:REM TASTO NON VALID
     O                         :rem   5
1810 GOSUB52000:GOTO1100
     ***** VALUTAZIONI A 16 BIT *****
                               :rem 247
2000 DV=1:IFKK=PTHENDV=2        :rem 238
2010 IFKK=QTHENDV=12            :rem 223
2015 REM DV:1=FREQ. VOCE, 2=LARGHEZZA IMP
     ULSO                      :rem 227
2016 REM 12=FREQ. FILTRO, CANCELLA PRECED
     ENTE                      :rem 183

2017 REM VARIABILI UTILIZZATE PER MAGGIOR
     E VELOCITA'               :rem   6
2018 REM ZZ=VAL MAX, H=REG SID  :rem 216
2020 GOSUB53020:XX=256:AA=0:BB=1:CC=2
                               :rem 134
2030 K=29:H=0:ZZ=65535:J=7      :rem 210
2040 IFDV=2THENZZ=4095:H=2      :rem  65
2050 IFDV=12THENZZ=2047:H=21:J=0:rem 144
2055 REM PRE-CALCOLA BUFFER E SID
                               :rem 146
2056 REM PUNTI DI INGRESSO      :rem  56
2060 YY=SID+EL*J+H:H=SBUF+EL*J+H:rem 204
2065 REM CALCOLA E VISUALIZZA VALORI
                               :rem 196
2066 REM SE VIENE TROVATO UN TASTO NON VA
     LIDO                      :rem 200
2067 REM IN UNA SUBROUTINE, ESCE CON IL V
     ALORE                     :rem 200
2068 REM INTATTO               :rem 215
2070 GOSUB200:GOTO1110:REM
     ***** VALUTAZIONI AD 8 BIT *****
                               :rem 197
3000 DV=4:H=EL*7+5:J=240:K=16:REM ATTACK
                               :rem  53
3005 REM DECAY, SUSTAIN, RELEASE?
                               :rem 209
3010 IFKK=DTHENDV=5:J=15:K=1    :rem 191
3020 IFKK=STHENDV=6:H=EL*7+6    :rem 200
3030 IFKK=RTHENDV=7:H=EL*7+6:J=15:K=1
                               :rem 227
3035 REM RISONANZA O VOLUME?    :rem 202
3040 IFKK=NTHENDV=13:H=23       :rem   5
3050 IFKK=VTHENDV=14:H=24:J=15:K=1
                               :rem  42
3055 REM AZZERA, CALCOLA E VISUALIZZA I V
     ALORI                     :rem   6
3056 REM SE VIENE PREMUTO UN TASTO NON VA
     LIDO                      :rem 197
3057 REM RITORNA CON IL VALORE INTATTO
                               :rem  51
3060 GOSUB53020:GOSUB400:GOTO1110:REM
     *****CALCOLO DEI BYTE DI CONTROLLO**
     ***                       :rem 213
4000 DV=-1:GOSUB53020:REM AZZERA:rem 152
4005 REM SETTA I VALORI DI MASCHERA
                               :rem  21
4010 XX=4+EL*7:DV=3:J=240:REM FORMA D' ON
     DA                        :rem 190
4020 IFKK=GTHENDV=8:J=1:REM PORTA
                               :rem  67
4030 IFKK=ITHENDV=9:J=4:REM RING:rem 244
4040 IFKK=YTHENDV=10:J=2:REM SYNC
                               :rem  56
4050 K=J:IFJ=240THENK=16        :rem 159
4055 REM PRENDE IL VALORE DAL BUFFER
                               :rem  89
4056 REM SALTA SE FORMA D' ONDA :rem 194
4060 GOSUB700:IFDV=3THEN4110    :rem 169
4066 REM INTERRUTTORE PER IL VALORE
                               :rem 176
4070 Z=-(Z=0):CL=EL*5:CT=DV     :rem  69
4080 IFZ=1THENPRINT"{RVS}";:REM EVIDENZIA
                               :rem  45
4090 GOSUB53070:GOTO4190:REM VISUALIZZA
                               :rem 189
4100 REM                        :rem 169
4105 REM INCREMENTA 0->1,1->2,2->4,4->8,8
     ->0                       :rem  89
4110 Z=(Z*2-(Z=0))*-(Z<8)       :rem  31
4120 CT=DV:CL=EL*5:GOSUB930:REM MUOVE IL
     CURSORE                   :rem 227
4125 REM VISUALIZZA FORMA D' ONDA
                               :rem 196
4130 A$="-":IFZ=1THENA$="T      :rem 179
4140 IFZ=2THENA$="S             :rem 103
4150 IFZ=4THENA$="P             :rem 103
4160 IFZ=8THENA$="N             :rem 106
4170 PRINTA$                    :rem 190
4180 REM                        :rem 177
4190 GOSUB800:GOTO1100:REM
     ***** CALCOLO FILTRI E MODO *****
                               :rem  24
5000 DV=-1:GOSUB53020:REM AZZERA PRECEDEE
     NTE                       :rem 189
5010 DV=15:CT=DV:CL=0:REM MODO  :rem  46
5015 REM ASSEGNAZIONE FILTRO?   :rem  69
5020 IFKK=N1THENDV=16:H=1:GOTO5500
                               :rem  66
```

```
5030 IFKK=N2THENDV=17:H=2:GOTO5500
                                 :rem  70
5040 IFKK=N3THENDV=18:H=4:GOTO5500
                                 :rem  75
5050 IFKK=XTHENDV=19:H=8:GOTO5500
                                 :rem  40
5055 REM 3 OFF?                  :rem   0
5060 IFKK=TTHENDV=22:GOTO5200    :rem  37
5070 REM                         :rem 176
5075 REM INCREMENTO DEL MODO     :rem 173
5080 XX=24:J=112:K=16:GOSUB700   :rem 239
5085 REM INCREMENTA 0->1,1->2,2->4,4->0
                                 :rem  90
5086 REM SONO POSSIBILI EVENTUALI MODIFIC
     HE AL PROGRAMMA             :rem  12
5087 REM PER RENDERE DISPONIBILE PIU' DI
     UN MODO                     :rem  74
5088 REM ALLO STESSO TEMPO       :rem  71
5090 Z=(Z*2-(Z=0))*-(Z<4)        :rem  35
5100 GOSUB930:REM MUOVE IL CURSORE
                                 :rem  68
5110 A$=" -":IFZ=1THENA$="LP     :rem 250
5120 IFZ=2THENA$="BP             :rem 165
5130 IFZ=4THENA$="HP             :rem 174
5140 PRINTA$:GOTO5550:REM VISUALIZZA
                                 :rem  46
5150 REM                         :rem 175
5155 REM SPEGNE LA 3             :rem  54
5200 CT=DV:XX=24:J=128:K=1:GOSUB700
                                 :rem  98
5205 REM MASCHERA E INTERRUTTORE:rem 240
5210 ZZ=128*-((ZANDJ)=0):H=1:GOTO5520
                                 :rem  96
5220 REM                         :rem 173
5225 REM ASSEGNA FILTRO          :rem 132
5500 CT=DV:XX=23:J=15:K=1:GOSUB700
                                 :rem  47
5510 ZZ=H*-((ZANDH)=0):REM MASCHERA ED IN
     TERRUTTORE                  :rem 129
5520 IFZZ>0THENPRINT"{RVS}";     :rem 202
5530 GOSUB53090:REM VISUALIZZA   :rem 126
5540 Z=ZANDJ-HORZZ               :rem 166
5550 GOSUB800:GOTO1100:REM
     ***** INSERZIONE I/O*****   :rem 171
6000 DV=-1:GOSUB53020:REM AZZERA:rem 154
6005 REM DEFAULT SU OSC          :rem  65
6010 H=OS:CT=20:CL=0:IFKK=0THEN6030
                                 :rem 119
6015 REM INVILUPPO               :rem 118
6020 H=EN:CT=21                  :rem  81
6030 J=PEEK(H):REM LEGGE VALORE :rem  89
6035 REM INTERRUTTORE VALORE     :rem  50
6036 REM SIA  255 (OFF),0 (VOCE 1,
                                 :rem 102
6037 REM BASSO),1 (VOCE 1, ALTO), O 22
                                 :rem 156
6038 REM (FREQ. FILTRO, ALTA)    :rem 128
6040 J=-(J=0)-22*(J=1)-255*(J=22)
                                 :rem 121
6050 POKEH,J:GOSUB930:REM SALVA VALORE
                                 :rem 108
6055 REM CALCOLA IL DISPLAY      :rem  78
6060 J=1-(J>0)-(J>1)-(J>22)      :rem  91
6070 PRINTMID$("LHF-",J,1):GOTO1100
                                 :rem 252
6996 REM
     ***** TASTI FUNZIONE*****   :rem  89
7000 DV=-1:GOSUB53020:REM AZZERA:rem 155
7005 REM NON PUO' ARRIVARE A QUESTE LINEE
      A MENO CHE                 :rem 215
7006 REM SIA STATO PREMUTO IL TASTO SHIFT
                                 :rem 227
7007 REM F1=F2,F3=F4,F5=F6       :rem 154
7010 H=0:CT=8:J=1:K=1:REM DEFAULT
                                 :rem 203
7015 REM F1(F2)=INTERRUTTORE, F3(F4)=TUTT
     O OFF                       :rem  13
7020 IFKK=F1ORKK=F3THEN7100      :rem 134
7025 REMF5(F6)=ON QUINDI OFF     :rem 121
7026 REM Z=LEGGE VALORE, PONE A 1
                                 :rem  71
7027 REM METTE L'INTERRUTTORE SU 0
                                 :rem  53
7028 REM H SERVE PER REGOLARE Z :rem  20
7030 H=1:GOSUB7100               :rem   2
7040 H=2                         :rem 130
7095 REM PORTE VOCI A SID+4,11,18
                                 :rem  23
7100 FORXX=4TO18STEP7            :rem  79
7105 :REM SE F3, REGOLA PER SPEGNERE
                                 :rem  34
7110 :GOSUB700:IFKK=F3THENZ=1    :rem  38
7115 :REM REGOLA PER F5          :rem   8
7120 :IFH>0THENZ=H-1             :rem 181
7125 :REM CALCOLA RIGA DI VISUALIZZAZIONE
                                 :rem  35
7130 :CL=-5*(XX=11)-10*(XX=18)   :rem  92
7135 :REM VISUALIZZA SE ON/OFF   :rem  63
7140 :ON(Z<>1)+2GOSUB53060,53070:rem 255
7150 :Z=-(Z=0):YY=YYAND254ORZ    :rem  39
7155 :REM SALVA NEL BUFFER       :rem   0
7160 :POKESBUF+XX,YY             :rem  32
7170 NEXT                        :rem  14
7175 REM PASSA VELOCEMENTE AL SID
                                 :rem 212
7180 FORXX=4TO18STEP7            :rem  87
7190 :POKESID+XX,PEEK(SBUF+XX)   :rem 162
7200 NEXT:RETURN:REM
     ** INIZIALIZZA **           :rem  54
49995 REM * INSERISCE LE ROUTINES DI SIST
      EMA  *                     :rem 158
49996 REM * ROUTINE DI PLOTTAGGIO CURSORE
      *                          :rem  29
50000 MOVE=679:CX=251:CY=252     :rem 101
50010 FORXX=MOVETOMOVE+7         :rem  61
50020 :READYY:POKEXX,YY          :rem 246
50030 NEXT:REM
      ROUTINE DI INTERRUPT       :rem 213
50040 FORXX=712TO760             :rem 167
50050 :READYY:POKEXX,YY          :rem 249
50060 NEXT:REM
      ***** ASSEGNAMENTO VARIABILI *****
                                 :rem  20
50100 KEY=203:SHFT=653:NULL=64   :rem  23
50105 REM PEEK(KEY)=VALORE TASTO CORRENTE
                                 :rem  49
50106 REM PREMUTO, PEEK(SHFT)>0 SE
                                 :rem 233
50107 REM SHIFT,C=,O CTRL SONO PREMUTI
                                 :rem  32
50108 REM PEEK(KEY)=NULLO SIGNIFICA NESSU
      N TASTO PREMUTO            :rem  40
50110 UP=7:FA=2:LO=1:XX=0:YY=0:ZZ=0:Z=0
                                 :rem 143
50115 REM UP=CURSORE SU/GIU'=LENTO
                                 :rem 135
50116 REM FA=CRSR SIN/DES=MODERATO
                                 :rem 108
50117 REM LO=RETURN=VELOCE       :rem 149
```

```
50118 REM NO SHIFT=GIU', SHIFT=SU
                                    :rem  214
50120 KK=0:HH=0:H=0:J=0:K=0:DV=0:CT=0
                                    :rem  131
50125 REM KK CONTIENE IL VALORE DEL TASTO
       CORRENTE                     :rem  236
50126 REM HH>0 SE SHIFT E' PREMUTO
                                    :rem  142
50127 REM DV,CT,CL SONO I VALORI DEL DISP
      LAY                           :rem   59
50130 CL=0:AA=0:BB=0:CC=0:SID=54272
                                    :rem  209
50135 REM OP=MUOVE INTERRUZIONE,CS=RIPRIS
      TINA                          :rem  252
50140 SBUF=MOVE+8:OP=712:CS=725 :rem   92
50145 REM VALORI TASTI EVIDENZIATI
                                    :rem  113
50150 F1=4:F3=5:F5=6:F7=3           :rem   87
50160 F=21:P=41:W=9:A=10:D=18:S=13:R=17
                                    :rem  177
50170 G=26:I=33:Y=25:Q=62:N=39:M=36
                                    :rem  212
50180 V=31:N1=56:N2=59:N3=8         :rem  235
50185 REM EN,OS USATI NELL' I/O :rem  231
50190 X=23:O=38:E=14:T=28:EN=254:OS=253
                                    :rem  216
50196 REM
      ***** LEGGE VALORI SCHERMO *****
                                    :rem  207
50200 DIMWX%(22),WY%(22):REM PAROLA X,Y
                                    :rem  202
50210 DIMVX%(22),VY%(22):REM VALORE X,Y
                                    :rem  211
50220 DIMWRD$(22):REM VISUALIZZA STRINGHE
                                    :rem   45
50230 FORXX=0TO22                   :rem    5
50240 :READWX%(XX),WY%(XX)          :rem   40
50250 :READVX%(XX),VY%(XX)          :rem   39
50260 :READWRD$(XX)                 :rem  101
50270 NEXT:REM
      ***** VISUALIZZA SCHERMO *****
                                    :rem   34
50300 PRINT"{CLR}{OFF} SID EDITOR - PREME
      RE LETTERE BIANCHE"           :rem  209
50310 PRINT"CRSR:[S/G]=LENTO,[S/D]=MED,[R
      ETURN]=VEL"                   :rem   35
50320 FORXX=0TO10STEP5:REM VOCI 1-3
                                    :rem   83
50325 :REM 0=F1,5=F3,10=F5          :rem   93
50330 :A$=RIGHT$(STR$(2*XX/5+1),1)
                                    :rem   97
50335 :REM MUOVE CURSORE            :rem  205
50340 :CT=0:CL=XX:OFS=-3:GOSUB900
                                    :rem    2
50345 :REM STAMPA "VOCE"            :rem   86
50350 :PRINT"{WHT}F"A$"[<7>] "WRD$(0)XX/5
      +1                            :rem  152
50355 :REM Z DETERMINEA IL TIPO DI VISUAL
      IZZAZIONE                     :rem   77
50360 :FORYY=1TO10:Z=1-(YY<8)-(YY=3)
                                    :rem  137
50370 :CT=YY:OFS=0:GOSUB900         :rem  161
50380 :PRINTWRD$(YY)" "MID$(" 0-",Z,1)
                                    :rem  242
50390 :NEXT                         :rem  122
50400 NEXT                          :rem   56
50410 REM                           :rem  222
50415 REM REGOLAZIONI FILTRI E VOLUME
                                    :rem   13
50420 CT=11:CL=0:GOSUB900           :rem  186
50430 PRINTWRD$(11)                 :rem   77
50440 FORXX=12TO22:IFXX<>16THEN50460
                                    :rem  195
50445 :REM PROMPT ADDIZIONALE       :rem   38
50450 :PRINT"{ 12 SPAZI}ASSEGNA FILTRO
                                    :rem  185
50455 :REM A$=VISUALIZZAZIONE DI DEFAULT
                                    :rem  236
50460 :A$="":IFXX<15THENA$=" 0      :rem   93
50470 :IFXX=15ORXX=20ORXX=21THENA$=" -
                                    :rem   29
50480 :CT=XX:GOSUB900:REM MUOVE CURSORE
                                    :rem  223
50490 :PRINTWRD$(XX)A$              :rem   64
50500 NEXT:PRINT                    :rem    0
50510 PRINT"{WHT}F2[<7>]=INTERR {WHT}F4
      [<7>]=TUTTO OFF {WHT}F6[<7>]=ON/OFF
       {WHT}F8[<7>]=END";           :rem   48
50515 REM VISULIZZA VALORI DI DEFAULT
                                    :rem   20
50520 EL=0:OEL=EL:DV=-1:ODV=DV      :rem   75
50598 REM
      *****INSERISCE I/O NELL'INTERRUPT *
      ****                          :rem   30
50600 POKEEN,255:POKEOS,255:SYSOP
                                    :rem   48
50996 REM
      ***** AZZERA SID E BUFFER *****
                                    :rem   65
51000 FORXX=SIDTOSID+28             :rem  194
51020 :  POKEXX,0                   :rem  109
51030 :  POKESBUF+XX-SID,0          :rem  214
51040 NEXT                          :rem   57
51050 RETURN:REM
      ***** BUZZ DI ERRORE **** :rem   26
52000 POKESID+23,0:REM NESSUN FILTRO
                                    :rem  188
52010 POKESID+24,15:REM MASSIMO VOLUME
                                    :rem   57
52020 POKESID+1,10:REM VOCE 1       :rem  109
52030 POKESID+5,0:REM NESSUN ATTACK/DECAY
                                    :rem   12
52040 POKESID+6,240:REM MASSIMO SUSTAIN
                                    :rem  139
52050 POKESID+4,0:REM PORTA DISATTIVATA
                                    :rem  168
52060 POKESID+4,33:REM DENTE DI SEGA
                                    :rem   56
52070 FORXX=1TO80:NEXT:REM RITARDO
                                    :rem  186
52075 REM RIPRISTINA VALORI PRECEDENTI
                                    :rem  156
52080 POKESID+23,PEEK(SBUF+23)      :rem    0
52090 POKESID+24,PEEK(SBUF+24)      :rem    3
52100 POKESID+1,PEEK(SBUF+1)        :rem  145
52110 POKESID+5,PEEK(SBUF+5)        :rem  154
52120 POKESID+6,PEEK(SBUF+6)        :rem  157
52130 POKESID+4,PEEK(SBUF+4)        :rem  154
52140 RETURN:REM
      ***** EVIDENZIA OPZIONE SCELTA ****
      *                             :rem   29
52995 REM LINEA 53000 AZZERA VOCE PRECEDE
      NTE                           :rem   44
52996 REM LINEA 53010 EVIDENZIA LA NUOVA
      VOCE                          :rem   55
52997 REM LINEA 53020 AZZERA OPZIONE PREC
      EDENTE                        :rem   39
52998 REM LINEA 53040 EVIDENZIA OPZIONE C
      ORRENTE                       :rem  127
52999 REM                           :rem  246
```

```
53000 CT=0:CL=5*OEL*-(CT<11):GOSUB53070
                                  :rem  215
53005 REM CL=RIGA CORRENTE        :rem   50
53010 CL=5*EL*-(DV<11):GOSUB53060
                                  :rem   77
53020 CT=ODV:CL=5*OEL*-(CT<11)    :rem  217
53030 GOSUB53070                  :rem  122
53040 OEL=EL:ODV=DV               :rem  164
53050 CT=ODV:CL=5*OEL*-(CT<11)    :rem  220
53060 PRINT"{RVS}";               :rem   28
53065 REM NESSUNA MODIFICA PER QUESTI VAL
      ORI                         :rem  207
53066 REM PORTA, ANELLO, ETC      :rem   93
53070 IFCT<0ORCT>14THEN53110      :rem  149
53080 IFCT=3ORCT=11THEN53110      :rem  150
53090 GOSUB900:REM PLOTTAGGIO     :rem   50
53100 PRINT WRD$(CT)              :rem  127
53110 PRINT"{OFF}";:RETURN:REM
         ***** DATI PROGRAMMA *****
                                  :rem   60
59998 REM ROUTINE DI MOVIMENTO CURSORE
                                  :rem  144
59999 REM                         :rem  253
60000 DATA 24,166,251,164,252,76,240,255
                                  :rem  178
60096 REM                         :rem  233
60097 REM                         :rem  234
60098 REM ROUTINE DI INTERRUZIONE
                                  :rem   76
60099 REM                         :rem  236
60100 DATA 120,169,226,141,20,3,169
                                  :rem  177
60110 DATA 2,141,21,3,88,96,120,169
                                  :rem  182
60120 DATA 49,141,20,3,169,234,141
                                  :rem  130
60130 DATA 21,3,88,96,166,253,48,6
                                  :rem  150
60140 DATA 173,27,212,157,0,212,166
                                  :rem  181
60150 DATA 254,48,6,173,28,212,157
                                  :rem  146
60160 DATA 0,212,76,49,234        :rem  255
62992 REM                         :rem  240
62993 REM                         :rem  241
62994 REM VISUALIZZAZIONE DATI SCHERMO
                                  :rem  188
62995 REM                         :rem  243
62996 REM INIZIO PAROLA X, INIZIO PAROLA
      Y,                          :rem   31
62997 REM INIZIO VALORE X, INIZIO VALORE
      Y,                          :rem   52
62998 REM PAROLA                  :rem  181
62999 REM                         :rem  247
63000 DATA 3,4,3,4,VOCE           :rem  190
63010 DATA 3,13,3,23,"{WHT}F[<7>]REQUENZA
      :                           :rem  159
63020 DATA 4,13,4,23,"LARGH {WHT}P[<7>]LS
      :                           :rem   78
63030 DATA 5,14,5,24,"({WHT}W[<7>]) ONDA:
                                  :rem  192
63040 DATA 3,30,3,37,"{WHT}A[<7>]TTACK:
                                  :rem  173
63050 DATA 4,31,4,37,"{WHT}D[<7>]ECAY:
                                  :rem   95
63060 DATA 5,29,5,37,"{WHT}S[<7>]USTAIN:
                                  :rem   42
63070 DATA 6,29,6,37,"{WHT}R[<7>]ELEASE:
                                  :rem    7
63080 DATA 6,13,6,13,"{WHT}G[<7>]ATE
                                  :rem  225
63090 DATA 6,18,6,18,"R{WHT}I[<7>]NG
                                  :rem  251
63100 DATA 6,23,6,23,"S{WHT}Y[<7>]NC
                                  :rem  248
63110 DATA 19,1,19,1,FILTRO/OUT   :rem  242
63120 DATA 19,13,19,23,"FRE{WHT}Q[<7>]UEN
      ZA:                         :rem   15
63130 DATA 20,13,20,23,"RISO{WHT}N[<7>]AN
      ZA:                         :rem    4
63140 DATA 19,30,19,37,"{WHT}V[<7>]OLUME:
                                  :rem   60
63150 DATA 20,32,20,37,"{WHT}M[<7>]ODO:
                                  :rem  134
63160 DATA 21,27,21,27,"{WHT}1[<7>]
                                  :rem   84
63170 DATA 21,29,2,29,"{WHT}2[<7>]
                                  :rem   41
63180 DATA 21,31,21,31,"{WHT}3[<7>]
                                  :rem   78
63190 DATA 21,33,21,33,"E{WHT}X[<7>]T
                                  :rem   17
63200 DATA 22,13,22,18,"{WHT}O[<7>]SC3
                                  :rem   51
63210 DATA 22,21,22,26,"{WHT}E[<7>]NV3
                                  :rem   54
63220 DATA 22,29,22,29,"DISA{WHT}B[<7>]
      { 3 SPAZI}3                 :rem  188
63230 REM
      *** FINE PROGRAMMA ***      :rem  166
```



PRINT

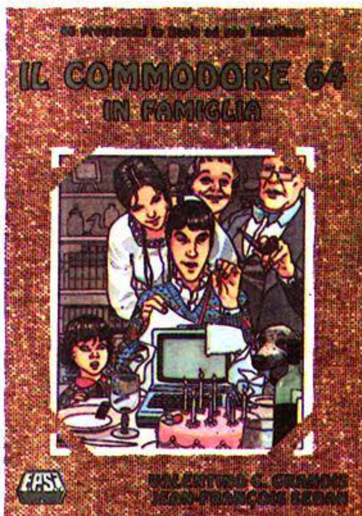
Recensioni Libri

di **M. Cristuib Grizzi**

I libri italiani vengono offerti con il 10% di sconto sul prezzo di copertina.

IL COMMODORE 64 IN FAMIGLIA

di **V.C. Grandis, J.F. Sehan**
Ed. **EPSI**

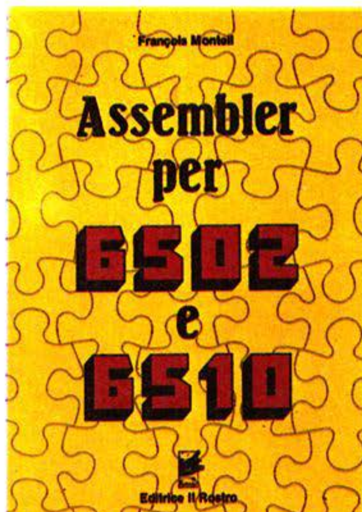


Si sente molto parlare delle possibilità di utilizzo del proprio computer per la gestione dei piccoli problemi che si possono presentare nella vita casalinga, ma le idee in merito sono sempre state piuttosto vaghe ed imprecise.
Questo libro propone finalmente dei programmi pratici, per tutte le esigenze veramente "home" che possono essere affrontate dal vostro C64. Si spazia infatti – nell'arco di ben quaranta programmi – attraverso otto diversi argomenti: finanze, pedagogia, cucina e maglieria, giochi nazionali, tempo, forma fisica, fai da te ed utilità generali.
Particolarmente interessanti la sezione finanze, dove vengono proposti programmi per la gestione del proprio conto corrente bancario, di un deposito a risparmio, del costo annuo della propria autovettura e del calcolo dell'IRPEF, e la sezione pedagogia, in cui vengono presentati programmi educativi per bambini che debbono imparare l'alfabeto, i verbi o i primi rudimenti di aritmetica.
Per la gioia delle massaie, nella sezione cucina e maglieria vengono proposti programmi di archivio ricette, menù casuali, gestione del congelatore e lavoro a maglia. Avrete così l'occasione – del tutto unica – di indossare maglioni e calzini computerizzati...
Per tutti i programmi vengono presentati i diagrammi di flusso, al fine di facilitarne la comprensione, mentre i listati sono ottenuti con stampante Commodore a matrice di punti e contengono quindi tutti i caratteri grafici e di controllo esattamente così come appaiono sul video.

ASSEMBLER PER 6502 E 6510

di **F. Monteil**
Ed. **Il Rosto**

Sono sempre più numerosi i possessori di C64 che, avendo ormai sviluppato una buona conoscenza della programmazione in BASIC, desiderano sperimentare quella in linguaggio macchina. Questo libro viene loro in soccorso, presentando una trattazione completa sul linguaggio ASSEMBLY del diffuso microprocessore 6502 e del 6510 implementato sul C64, che, dal punto di vista istruzioni ASSEMBLY, si presenta assolutamente identico al primo.
La prima parte consta delle introduzioni di rito sul sistema di numerazione binario e BCD, codifica ASCII e sistema di numerazione esadecimale. Si passa quindi all'analisi della sintassi richiesta dal linguaggio ASSEMBLY, della procedura di assemblaggio e di alcuni degli ASSEMBLER correntemente disponibili per questi due microprocessori.
Il set di istruzioni viene suddiviso per argomenti, analizzando separatamente ogni gruppo di istruzioni in funzione dell'operazione svolta, ed un capitolo separato viene riservato alle linee di input/output del 6510.
Notevole rilievo viene conferito alle procedure di "debug", particolarmente impegnative per i programmi in codice macchina, ed un capitolo tratta di alcune istruzioni "misteriose" del 6502, presenti sotto diversa forma pure nel 6510 e dipendenti anche dall'anno di realizzazione del microprocessore stesso.
Un testo completo, anche se piuttosto tecnico, adatto soprattutto a programmatori esperti della struttura di memoria del C64.



I libri recensiti possono essere ordinati utilizzando il coupon di pag. 50

input·output

Piccoli annunci

Vendo, cambio programmi per C64. Giochi in LM (oltre 130) e molte utility (es. Turbotape, Turbodisk, Magic Desk, ecc.) Lista gratuita. Lello Lavacca - Via P. Cioffrese, 6 - 70092 Bitonto (BA)

Vendo, cambio centinaia di programmi per C64: videogiochi, utility e varie. Tutto in linguaggio macchina ed a prezzi modicissimi. Su richiesta invio lista dettagliata. Roberto Ferraro - Vico Buono, 7 - 80069 Vico Equense (NA) - Tel. 081/8799937

Per CBM 64 cambio, vendo circa 300 programmi (utilities - giochi) preferibilmente su disco. Rispondo a tutti. Annuncio sempre valido. Scrivere o telefonare a: Salvatore Intilisano - Via R. Margherita, 113 - 98028 S.Teresa Riva (ME) - Tel. 0942/793167

Per passaggio a sistema superiore vendo a L. 300.000 trattabili VIC 20 nuovo completo di manuale, libro giochi, cassette con programmi ed utilità, bilancio familiare, giochi vari. Telefonare ore pasti. Annalisa Pasetti - Via Mascheroni, 9 - 20051 Limbiate (MI) - Tel. 02/9963063

Cambio programmi di ogni genere su cassetta per VIC 20. Alberto Lupo - Via S.Francesco d'Assisi - 75024 Montescaglioso (MT) - Tel. 0835/207032

Vendo VIC 20 più registratore, introduzione al BASIC parti 1 e 2, joystick, 3 cartridge a L. 300.000. Roberto Caroti - Via B. Croce, 142 - 52100 Arezzo - Tel. 0575/322982

Vendo per VIC 20 4 eccezionali giochi: Road Race, Raid On Fort Knox, Jupiter Lander e Star Post, valore reale L. 170.000 a sole L. 50.000. Se siete interessati rivolgetevi a: Roberto Pinochi - Via Colmate del Cerro - 51019 Ponte Buggianese (PT) - Tel. 0572/634115

Vendo per C64 6 giochi più Turbo su cassetta a sole L. 10.000 più spese postali (Sky Race, Frogger, Calcio, Zaxxon, Scacchi, Decathlon, Atzec Challange, One On One, Flip And Flop, Hunck Back). Annuncio sempre valido. Marco Gamannossi - Via T. Cortesi, 19 - 50047 Prato (FI)

Cambio software su disco per CBM 64. Attendo lista. Preferisco programmi gestionali ed utility. Giovanni Pugliese - Via A. Volta, 93 - 74100 Taranto - Tel. 099/413769

Vendo VIC 20 più registratore ed espansione. Davide Ragaglini - Via Sforza, 7 - 54031 Avenza (MS) - Tel. 0585/58326

Vendo VIC 20 più registratore C2N, Hes Mon, Super Expander, Scacchi, Gorf, 4 libri, molti giochi su cassetta. Tutto a L. 300.000. Roberto Stanghellini - Via Italia, 40 - 20094 Corsico (MI) - Tel. 02/4583257

Vendo registratore della Commodore a causa del decesso prematuro del mio C64, regalando una serie di cassette originali. Inviare offerta. Mauro Polidori - Via Fraz. Capanna, 1 - 42035 Castelnuovo Monti (RE)

Cambio, vendo per C64 Basket, Mundial Soccer, Ghostbusters, Decathlon, Popeye, Burger Time; tutti in Turbo Tape, solo su cassetta. Preferibilmente scambio solo nella mia provincia. Invierò la mia lista a chi invierà la sua. Pinuccio Moccagatta - Via Paolo Bocca, 17 - 15010 Rivalta Bormida (AL) - Tel. 72124

Vendo cassetta con 3 utility e 3 giochi per CBM 64 a L. 20.000, incluse spese di spedizione. Accetto telefonate dalle 14.00 alle 16.30. Domenico Pastore - Via Municipio, 11 - 82030 Pietraroia (BN) - Tel. 0824/862228

Eccezionale! Regalo cartucce per VIC 20 a chi acquista tutta la mia serie di programmi (circa 300) al prezzo di L. 85.000 regalo joystick a chi acquista programmi per CBM 64 (circa 150) al prezzo di L. 45.000. Vendo anche separatamente. Luciano Zenere - Via Gorizia - 36010 Monticello Conte Otto (VI) - Tel. 0444/596079

Vendo VIC 20 più cartridge Super Expander, 2 libri, moltissimi programmi a L. 250.000. Sono a disposizione di chi voglia provare il sistema. Antonio Di Palma - Via A. Petrosini, 10 - 84014 Nocera Inferiore (SA) - Tel. 081/923810

Vendo, cambio giochi per CBM 64. Disponibili 500 giochi tra cui Calcio (con replay), Bruce Lee, Summer Games, Strip, Poker, ecc. Venti giochi L. 40.000 e 4 giochi in omaggio. Telefonare dalle 14.30 in poi. Pietro Micarelli - Via Degli Scipioni, 40 - 00192 Roma - Tel. 06/314510

Cerco "Commodore Computer Club" nr. 1 e 2 (anche solo fotocopie). Telefonare ore serali. Alberto Bacchilega - Via Bellini, 36 - 40026 Imola (BO) - Tel. 0542/41941

Vendo per C64 i più richiesti e selezionati video games oggi sul mercato, a prezzi irrisori. Esempio: Arabian Nights, Mission Impossible, Raid Over Moscow, ecc, a L. 2.000 (solo su cassetta). Inviate le vostre richieste. Maurizio Anfosso - Via Simonetti, 4 - 14018 Roatto d'Asti (AT) - Tel. 0141/938118

Cambio giochi per VIC 20 con o senza espansione da 3 Kbyte. Per informazioni telefonare. Michele Tampieri - Via E. Farini, 27 - 48020 San Patrizio in Vincoli (RA) - Tel. 0544/551018

Vendo cambio per CBM 64 500 giochi tra cui: Football Manager, Summer Games, Match Point, Impossible Mission, ecc.; i programmi sono su cassetta. Venti giochi L. 40.000 con 4 in omaggio (cassetta compresa nel prezzo). Telefonare ore pasti. Eriberto Arciulo - Via Ottaviano, 73 - 00192 Roma - Tel. 06/386135

Per Commodore 64 vendo cassette con 5 giochi a L. 4.000. Sconto per acquisto di più cassette. Vendo anche Calcio e Tennis per C64 e programmi per ZX Spectrum. Maurizio Montorsi - Via P. Giacometti, 1/7 - 16100 Genova - Tel. 010/513074

Per C64 vendo programmi di vario genere da L. 3.000 a L. 10.000. Qualche esempio: Stat 64, Maxi Archivio, Mission Impossible, Ghostbusters, Raid Over Moscow, registrati con Turbo Tape. Elenco gratuito. Stefano Longhi - Via Verdi, 25 - 20053 Muggiò (MI)

Compro, vendo, cambio programmi per C64. Posseggo oltre 150 giochi ed utilità. Inviare le liste con prezzi e descrizioni. Rispondo a tutti. Umberto Bettini - Via Marconi, 14 - 40033 Casalecchio (BO) - Tel. 575276

Vendo e cambio software per C64. Dispongo di molti giochi su disco e cassetta (Ghostbusters, Pac Man 3D, Pit Stop II, ecc.). Richiedere la lista a: Marco Carnevale - Via Di Porta a Mare, 19 - 56100 Pisa - Tel. 050/20974

Cerco per C64 programmi di utility in cui si utilizza il joystick anziché la tastiera, come Magic Desk, Magic Hand, e così via. Inoltre cerco anche programmi riguardanti la grafica e il disegno strutturati in modo analogo. Eluis Spagnolatti - Via Monastero - 23010 Berbenno (SO) - Tel. 493308

Carissimi commodoristi, io vendo un VIC 20 con joystick e circa 120 giochi a L. 199.900. Telefonare ore pasti. Massimo Sandrelli - Via Angelo Olivieri - 00122 Lido di Roma (RM) - Tel. 06/5600865

Per CBM 64 vendo centinaia di programmi: video games, utility, gestionali con manuali, e tanti altri. Per avere la lista rivolgersi a: Stefano Dominioni - Via Tommaseo, 18 - 21100 Varese - Tel. 0332/229909

Vendo VIC 20 più registratore, cartuccia "Super Smash", paddle, joystick per un valore di L. 491.000 a L. 310.000. Oppure cambio con ZX Spectrum 16 Kbyte. Regalo cassette. Fabio Terenziani - Via Gioacchino Rossini, 108 - 05100 Terni - Tel. 0744/86396

Vendo VIC 20 più registratore, joystick, 100 programmi tra giochi ed utility, cartridge Sargon II (fantastico gioco di scacchi). Occasionissima! Solo L. 300.000 trattabilissime. Telefonare ore pasti. Emmanuel Latilla - Via Livorno, 45 - 00162 Roma - Tel. 06/423081

Per C64 cambio programmi (giochi ed utilities). Inviare lista a: Alfonso Bolognini - Via Savoia, 15 - 07100 Sassari - Tel. 079/291353

Vendo C64 più stampante MPS801, registratore C2N, joystick e oltre 30 giochi, tutto nuovissimo, a L. 1.000.000. Inoltre regalo pacco da 2.000 fogli per stampante, alcuni volumi ed, eventualmente, anche TV b/n. Ugo Tanfani - Via V. Monti, 1 - 20090 Trezzano (MI) - Tel. 4455748

Cambio software per C64. Cerco software per light pen e di comunicazione con altri computer o periferiche non Commodore. Roberto Maugeri - Via V. Brancati, 6 - 95128 Catania - Tel. 095/432846

Per C64 cambio, compro, vendo software. Possiedo tantissime novità da tutto il mondo. Mandatemi le vostre liste, riceverete la mia. Marco De Fernex - Corso G. Matteotti, 39 - 10121 Torino - Tel. 011/511042

Vendo ottimi programmi per C64. Per maggiori informazioni scrivere a: Massimo Tabasso - Piazza Molineris, 1 - 12038 Savigliano

Per C64 vendo software stupendo, giochi, utility, ecc. Dispongo di vari programmi fra i quali: Jumpman Junior, QBert, B.C., Pitstop, Hero, Hunck Back (Il Gobbo), Calcio (Commodore), ecc.; utility fra cui Easy Script, Turbo Tape, Dum Isgi, ecc. Dispongo anche di software per VIC 20. Contattare: Andrea Traldi - Via Cosimo Del Fante, 8 - 20122 Milano - Tel. 02/8323340

Vendo VIC 20 completo di joystick, 50 programmi di gioco, 2 manuali di giochi da copiare, cassette di Supervic fino al nr. 6 a L. 230.000. Vincenzo Torrisi - Via Vito D'Anna, 16 - 95024 Acireale (CT) - Tel. 606766

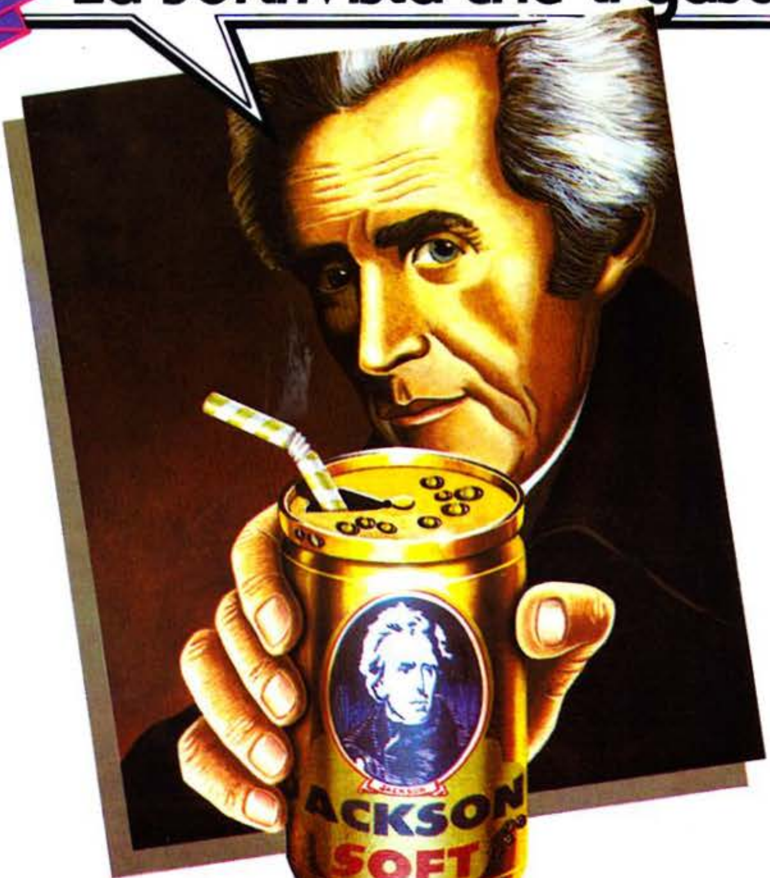
Compro una tastiera per VIC 20 a prezzo modico. Rispondo a tutti. Stefano Flammia - Via Fosso Cavallo, 91 - 83040 Fontanarosa (AV)

Vendo per VIC 20 programma su cassetta per sviluppo sistemi integrali del Totocalcio, Totip ed Enalotto a L. 30.000. Tengo a precisare che il programma è fra i più veloci che si possano trovare sul mercato. Massima serietà. Cerco inoltre urgentemente cartuccia Simons' BASIC per C64. Alberto Sizzardi - Via Lissa, 48 - 30021 Duma Verde di Caorle (VE) - Tel. 0421/83302

Vendo VIC 20 (meno di tre mesi di vita) ancora in imballo originale più registratore Commodore C2N, joystick, espansione di memoria 16 Kbyte, introduzione al BASIC parte 1 e parte 2, numerosi giochi espansi e non. Tutto a L. 550.000. Daniele Restelli - Viale Molise, 1 - 20092 Cinisello Balsamo (MI) - Tel. 02/6176956

FINALMENTE!

La Softrivista che ti gasa!



QUALCOSA DI SUPER, DI INEDITO, DI IRRESISTIBILE

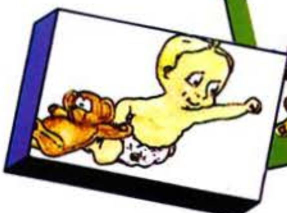
IL VERO GIOCO COMINCIA ADESSO

IN EDICOLA JACKSON SOFT SERIE ORO

I giochi esclusivi per Commodore 64 e Spectrum 48 K importati dall'Inghilterra, mai presentati in Italia.
Una sfida Jackson al già visto, al già fatto, al... già registrato.







continua la serie delle entusiasmanti avventure di **WALLY**

Corri in edicola, il vero gioco comincia solo adesso e se sei davvero bravo partecipa alla "sfida al campione", utilizzando il tagliando che troverai sull'ultima pagina di copertina di ogni numero.



GRUPPO EDITORIALE JACKSON